

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

**System pro správu organizace
konference**

System for conference management

2011 Bc. Jaroslav Šeps

Prázdná stránka, která se pak vyhodí, aby se zde mohlo dát zadání

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 1. 5. 2011

.....

Poděkování

Rád bych poděkoval vedoucímu diplomové práce Ing. Michalu Radeckému za odbornou pomoc a konzultaci při vytváření této práce.

Abstrakt

Práce seznamuje čtenáře nejprve s problematikou konferenčních systémů. Následně se věnuje teorii agilní metodiky SCRUM. Dále popisuje konkrétní projekt, který byl vyvíjen za dodržování principů již zmiňované metodiky. Celý proces vývoje pomocí agilní metodiky SCRUM je v poslední části podrobně vysvětlen formou případové studie. Výsledný systém a použítá metodika jsou také zhodnoceny pomocí konkrétních výsledků naměřených v průběhu vývoje a ostrého provozu.

Klíčová slova

Konferenční systémy, agilní metodiky, softwarový proces, případová studie, SCRUM

Abstract

This paper introduces reader with the issue of conference systems at first. Subsequently, the theory deals with the SCRUM Agile methodology. It also describes a specific project, which was developed in compliance with the principles already mentioned methodology. The whole process of development through the SCRUM Agile methodology has recently explained in detail through case studies. The resulting system and the methodology are assessed using the specific results obtained during the development and online running.

Keywords

Conference systems, agile methods, software process, case study, SCRUM

Seznam použitých symbolů a zkratek

.NET	.NET Framework
ASP	Active Server Pages
C#	C Sharp (programovací jazyk)
CMS	Conference Management Systems
CSS	Cascading Style Sheets
IEEE	Institute of Electrical and Electronics Engineers
IIS	Internet Information Server
LINQ	Language Integrated Query
MSSQL	Microsoft SQL
MVC	Model-View-Controller
MySQL	My Structured Query Language
OSA	Optical Society
PHP	Hypertext Preprocessor
SPIE	Society of Photographic Instrumentation Engineers
SŘBD	Systém řízení báze dat
WWW	World Wide Web

Seznam tabulek

Tabulka 1: Srovnání současných konferenčních systémů	16
Tabulka 2: Systém vyhodnocení stavu článku v konferenčním systému RTT	33
Tabulka 3: Product Backlog	40
Tabulka 4: Release plan	42
Tabulka 5: Task	43
Tabulka 6: Sprint Backlog #1	45
Tabulka 7: Sprint Backlog #2	47
Tabulka 8: Sprint Backlog #3	50
Tabulka 9: Sprint Backlog #4	51
Tabulka 10: Velocita týmu	53

Seznam obrázků

Obrázek 1: Use Case diagram	29
Obrázek 2: Architektonický vzor MVC	30
Obrázek 3: Administrace uživatelů v konferenčním systému RTT	31
Obrázek 4: Správa příspěvků zaslaných do konferenčního systému RTT	32
Obrázek 5: Recenzní formulář v konferenčním systému RTT	33
Obrázek 6: Stavový diagram životního cyklu článku v konferenčním systému RTT	34
Obrázek 7: Graf přístupů RTT 2010	35
Obrázek 8: Tým	38
Obrázek 9: Graf odhadu pro 1. Release	41
Obrázek 10: Scrum	42
Obrázek 11: Task Board.....	44
Obrázek 12: Burn-Down Chart Sprint #1.....	46
Obrázek 13: Burn-Down Chart Sprint #2.....	48
Obrázek 14: Burn-Down Chart Sprint #3.....	49
Obrázek 15: Burn-Down Chart Sprint #4.....	52
Obrázek 16: Graf výkonnosti vývojářů.....	53
Obrázek 17: Burn-Down Chart Odhad vs. Realita	54

Obsah

1	Úvod.....	11
2	Konferenční systém.....	12
2.1	Typické vlastnosti.....	12
2.1.1	Uživatelské role	12
2.1.2	Fáze.....	12
2.2	Moderní trendy	13
2.3	Současné konferenční systémy	14
2.3.1	Open Conference Systems.....	14
2.3.2	EasyChair	14
2.3.3	OpenConf	15
2.3.4	Conference Review System.....	15
2.4	Porovnání současných systémů	15
3	SCRUM.....	17
3.1	Pojmy.....	17
3.1.1	Scrum	17
3.1.2	Sprint.....	17
3.1.3	Plánovací poker	17
3.1.4	Release	18
3.2	Role.....	18
3.2.1	Vlastník produktu (Product owner).....	18
3.2.2	Scrum master	18
3.2.3	Scrum tým	18
3.2.4	Manažeri.....	19
3.2.5	Stakeholdeři.....	19
3.3	Artefakty	19
3.3.1	Product Backlog.....	19
3.3.2	Sprint Backlog	19
3.3.3	Burn-down chart	19
3.4	Scrum schůzky	20
3.4.1	Daily Scrum Meeting.....	20
3.4.2	Sprint Planning Meeting.....	20
3.4.3	Sprint Review Meeting	20
3.4.4	Sprint Retrospective.....	20

3.5	Fáze vývoje	20
3.5.1	Předehra.....	20
3.5.2	Hra	21
3.5.3	Dohra.....	21
4	Systém RTT	22
4.1	Ukázkový projekt	22
4.2	Specifikace požadavků	22
4.2.1	Funkční požadavky	22
4.2.2	Vstupy do systému	24
4.2.3	Výstupy ze systému	26
4.2.4	Vybrané funkce systému	27
4.2.5	Nefunkční požadavky.....	28
4.2.6	Diagram případů užití	28
4.3	ASP.NET MVC Framework	29
4.4	Nasazení do provozu.....	30
4.5	Ukázky aplikace	31
4.6	Statistiky systému	34
4.6.1	Google Analytics	35
5	Případová studie SCRUM	37
5.1	Proč Scrum?	37
5.2	Použití Scrum	37
5.2.1	Tým.....	37
5.2.2	Předehra.....	38
5.2.3	Hra	42
5.2.4	Dohra.....	52
5.3	Velocita týmu	53
5.4	Hodnocení metodiky.....	54
6	Závěr	55

1 Úvod

V dnešní době je neoddělitelnou součástí univerzit také publikační činnost. Tato činnost se postupně stala jedním z hlavních kritérií hodnocení kvality těchto univerzit. Vysoké školy dnes vedou své studenty k publikační činnosti, pořádání konferencí a vydávání sborníků. Zajištění těchto činností však není jednoduchá a vyžaduje spousty úsilí. Smyslem konferenčních systémů, kterým se věnuje tato práce, je podpora činností spojených s přípravou konferencí. Jedním z hlavních cílů této práce je vytvoření právě takového konferenčního systému.

V posledních letech se také v oblasti řízení projektů hovoří o agilních metodikách. Upouští se tedy od tradičních metodik, jejichž nedostatkem byla jejich složitost, malá nebo skoro žádná flexibilita čímž nestačily reagovat na změny požadavků nebo na nové požadavky a také byly nevhodné pro příliš malé týmy kvůli své robustní dokumentaci. Ani postupné vylepšování těchto metodik nedovedlo vymítnout tyto nedostatky. Cílem této práce je mimo jiné také představení jedné vybrané agilní metodiky v oblasti vývoje software a její význam v oblasti projektového řízení. Agilní metodiky nejsou podrobné, dbají na rychlý vývoj, jsou iterativní, inkrementální a dbají na komunikaci nejen se zákazníkem ale také uvnitř vývojového týmu. Vycházejících metodik je v současné době velké množství a stále vznikají nové a nové metodiky, které vyhovují konkrétnímu autorovi. Tyto nové metodiky většinou vycházejí z již starších metodik a jsou pouze v některých částech buď upravené anebo kombinují více metodik zároveň.

První část této diplomové práce podává ucelený obraz problematiky týkající se konferenčních systémů pomocí porovnání v dnešní době nejpoužívanějších konferenčních systémů. Dále bude popsána agilní metodika, konkrétně SCRUM, která byla využita při vývoji reálného projektu – konferenčního systému, jež bude představen v kapitole následující. V poslední kapitole pak bude uvedena případová studie použití agilní metodiky SCRUM při vývoji konferenčního systému. V závěru práce bude uvedeno závěrečné zhodnocení použité metodiky při vývoji projektu.

2 Konferenční systém

S rozvojem internetu a nástupu webu se webová platforma ukázala jako vhodné místo pro vznik konferenčních systémů označovaných zkratkou CMS (Conference Management Systems). Konferenční systémy jsou webové aplikace, které ulehčují přípravu a organizaci vědeckých konferencí.

Tak jako se vyvíjel internet a webové aplikace, tak se vyvíjely i CMS systémy od prvních systémů, které podporovaly pouze emailovou komunikaci mezi účastníky až po dnešní komplexní systémy, které pomáhají při všech aspektech přípravy a organizace vědecké konference. Nástup CMS systémů byl poměrně rychlý a už od poloviny devadesátých let tvoří neodmyslitelnou součást organizace vědeckých konferencí. Momentálně je na trhu dostatek CMS systémů ať už komerčních nebo volně dostupných bezplatných alternativ.

Na konferenční systém nemusíme nahlížet jen jako na systém, jehož použití vede jen k přípravě vědecké konference. Může být použit např. při vydávání vědeckého časopisu nebo při přípravě sborníků. Konferenční systémy jsou vlastně typem programů, které slouží pro sběr elektronických dokumentů, které se uchovávají v elektronickém úložišti. Dokumenty jsou následně vyhodnocovány. Je umožněno vyhledávání a třídění dokumentů. V našem případě jsou tyto dokumenty články nebo konferenční příspěvky.

2.1 Typické vlastnosti

2.1.1 Uživatelské role

Uživatelské role, které v konferenčních systémech vystupují, se mohou lišit vzhledem k poskytované funkcionalitě programu. Většina z nich ale obsahuje základní role, které jsou následující.

- Organizační skupina (Program Committee Chair)
- Recenzenti (Program Committee Member)
- Autoři příspěvků

2.1.2 Fáze

Inicializační fáze zahrnuje nastavení konferenčního systému organizátory. V této části organizátoři nastavují informace o konferenci nebo sborníku. Systémy umožňují nastavení povolených témat konference, kterých se zaslané příspěvky musí týkat. Je to také část, ve které se nastavují hraniční termíny, které určují, do kdy je možné přidávat příspěvky, čas na vykonání recenzí atd. Některé systémy umožňují nastavení celkového vzhledu konferenčního systému, přidání konferenčního loga nebo změnu barev.

Zaregistrovaní autoři v přispívající fázi zasílají své příspěvky. Registrace autorů vyžaduje většinou kromě jména a emailu i informace o pracovišti nebo odborné činnosti. Některé systémy přijímají nejdříve abstrakt příspěvku, na základě kterého se určí recenzent a až poté

samotný text příspěvku. Autor při každém příspěvku určuje téma, kterého se příspěvek týká, abstrakt, klíčová slova nebo jiné vyžadované údaje. Podporované formáty bývají nejčastěji pdf, doc, docx, rtf.

Recenzní fázi můžeme rozdělit na dvě části: výběr recenzentů a hodnocení příspěvků. Přiřazení vhodných recenzentů k zaslaným článkům je náročná část. Je důležité, aby příspěvky hodnotili recenzenti, kteří jsou experty v dané oblasti, které se příspěvek týká. Recenzenti při registraci do konferenčního systému zadávají svou odbornou způsobilost a zároveň témata příspěvků, které by měli zájem recenzovat. Autoři při přidávání příspěvku zadávají témata příspěvku a abstrakt. Tyto informace se používají při výběru recenzenta k příspěvku. Počet recenzentů k jednomu zaslanému článku bývá většinou pevně daný a většinou se jedná o dva až čtyři recenzenty. Při výběru recenzenta je důležité vyhnout se střetu zájmů mezi recenzenty a autory. Nejzákladnější jsou tyto dva konflikty.

- Autoři mohou zasílat příspěvky do konference, ve které zároveň vystupují jako recenzenti
- Příspěvek může mít více autorů, čímž se zvyšuje možnost výskytu prvního konfliktu

Po přiřazení recenzentů k jednotlivým příspěvkům může začít samotné hodnocení příspěvků. Jednotlivé konference nebo redakční rady žurnálů mají často přesná pravidla a konferenční systémy umožňují některé tyto pravidla aplikovat. Konferenční systémy poskytují formuláře, které recenzent na základě odborného zhodnocení vyplní. Nejjednodušší formuláře umožňují hodnotit příspěvek celkovou známkou podobně jako ve škole. Pokročilejší formuláře umožňují hodnocení jednotlivých konkrétních okruhů příspěvku. Příspěvek vždy zhodnotí vícero recenzentů. Recenzent má i možnost textového vyjádření k příspěvku v podobě komentáře.

Autoři schválených příspěvků posílají jejich finální verze, upravené podle námitek recenzentů. Formát konečného příspěvku může být předepsaný podle publikačních pravidel jednotlivých sborníků. Na konci této fáze nastává buď samotná konference, nebo publikace schválených příspěvků.

2.2 Moderní trendy

Jednou z nejkomplikovanějších a zároveň nejdůležitějších částí přípravy kvalitní konference je způsob přidělování zaslaných článků konkrétním recenzentům. V současnosti se tato fáze považuje za úzké hrdlo konferenčních systémů. Tato činnost je časově náročná protože je většinou vykonávána ručně na základě preferencí recenzentů a podle jejich oboru působnosti. Do úvahy se musí brát i možné konflikty zájmů a při velkém počtu příspěvků a recenzentů je situace velmi složitá. Momentálně vývoj konferenčních systémů směřuje k odstranění těchto časově náročných částí snahou o zautomatizování procesů pomocí znalosti umělé inteligence.

Jedním z projektů snažících se o odstranění tohoto problému konferenčních systémů je projekt GRAPE (Global Review Assignment Processing Engine). GRAPE se snaží o eliminaci ručního

přidělování recenzenta k článkům, případně dovoluje ruční přidělování v nejasné situaci nebo při výskytu konfliktu. Více informací lze dohledat v [1].

2.3 Současné konferenční systémy

Nyní budou uvedeny v současné době nejspíše nejpoužívanější systémy pro správu konferencí, které lze vyhledat na internetu a využít jejich služeb. Představeny budou systémy: Open Conference Systems (dále již jen jako OCS), EasyChair, OpenConf a Conference Review Systém (dále již jen jako CRS). Samozřejmě existují i další řešení, která jsou dostupná na internetu například: Conference Online-Management Systém (COMS), Confious nebo ConfTool.

2.3.1 Open Conference Systems

OCS je bezplatný webový publikační nástroj, který vytvoří kompletní webové prezentace pro vědecké konference. Mimo jiné umožňuje vytvořit webové stránky konference, uspořádat a zasílat výzvy k nahrávání dokumentů do systému, elektronicky přijímat články a abstrakty článků, upravovat autorům své práce, registraci a následnou správu účastníků, integrovat post-konferenční on-line rozhovory. Jedná se o PHP webovou aplikaci využívající SŘBD buď MySQL nebo PostgreSQL a jako server využívá Apache nebo Microsoft IIS. Tento systém je tedy možné použít na libovolném operačním systému, který podporuje výše zmíněné aplikace. Jelikož se jedná o volně stažitelný software, je zde možnost libovolné modifikace podle GNU licence. S tím také souvisí fakt, že tato aplikace funguje bez jakékoliv záruky. V současnosti je možnost vyzkoušení druhé verze této aplikace. Na oficiálních stránkách je také uživatelská podpora, kde je možnost dohledat se rady s případnými potížemi. Od roku 2009 do dnešního dne používá tento systém již přes 1200 uživatelů. OCS je také vícejazyčný systém, který umožňuje publikovat konference v různých jazycích. Na podpoře pro další jazyky se podílí komunita kolem OCS. Tento systém umožňuje nejen konference publikovat v různých jazycích, ale také poskytuje vícejazyčnou podporu, takže je možné mít konference zveřejněny ve dvou nebo dokonce třech jazycích. Tento systém v současnosti podporuje 11 světových jazyků, avšak čeština se mezi nimi zatím nevyskytuje. Více informací o systému naleznete na [2].

2.3.2 EasyChair

Jedná se o bezplatný systém pro správu konferencí, který je flexibilní, jednoduchý k použití a má mnoho funkcí, které ho dělají přizpůsobivý na různé modely konferencí. V této chvíli je to pravděpodobně nejpoužívanější systém pro správu konferencí. Systém byl navržen tak aby pomáhal správcům konferencí vypořádat se, se složitým procesem rozhodování o člancích. Aktuální verze umožňuje například monitorování a řízení programového výboru, sofistikované a flexibilní řešení konfliktů zájmu přihlášených členů a recenzentů, automatické zasílání článku, seznam nejnovějších událostí, předložení hodnocení, zasílání e-mailů autorům, recenzentům a členům konference, on-line diskuze o člancích, automatickou přípravu sborníku a řadu dalších funkcí. Tento systém je připraven být flexibilní, aby se přizpůsobil danému projektu. Více informací o systému naleznete na [3]. Pomocí tohoto software bylo již uskutečněno přes 8800 konferencí, což je jistě velice vysoký počet a jednotlivé konference si můžete prohlédnout na

oficiálních stránkách software. Tento systém se neinstaluje přímo na server daného pořadatele, jelikož veškerá logika se děje přímo v oficiálním systému. Všechny konference jsou tedy centralizované přímo u výrobce.

2.3.3 OpenConf

Tento systém je další volně dostupných systémů na internetu avšak pouze v komunitní edici podléhající vlastní licenci. V aplikaci se dočkáte funkcí, jako jsou online recenzování článků, emailová upozornění, zabezpečení pomocí CAPTCHA při zasílání článků, vícejazyčnou podporu. Jedná se o systém naprogramovaný ve skriptovacím jazyce PHP za použití MySQL databázového systému. Systém je možné nainstalovat si buď na svůj vlastní server, nebo využít možnosti použít server od výrobců avšak tato možnost již není zadarmo. Dále se zde nachází edice professional, která nabízí navíc například vlastní server, technickou podporu, online programy, diskuzi komise, více typů zaslání článku atd. Na oficiálních stránkách projektu je možnost vyzkoušet si online demonstrativní verzi tohoto systému. I tento systém je hojně používán a počet konferencí a různých jiných událostí, které se pomocí tohoto software dají evidovat, přesáhl tisícových hodnot. OpenConf je používán po celém světě minimálně v 90 zemích. Co se týká více jazyčné podpory, je systém přeložen plně do angličtiny a americké angličtiny. Dále se pracuje na zjednodušené čínštině, francouzštině a připravuje se portugalština a španělština. Více informací o systému naleznete na [4]. V češtině tedy tento systém není a asi ani v blízké době nebude.

2.3.4 Conference Review System

CRS poskytuje komplexní online zaslací a hodnotící systém jak pro akademické tak i pro odborné konference. CRS umožňuje uživatelům, aby předložila doklady pouze prostřednictvím webového prohlížeče, zefektivňuje celý recenzní proces, a poskytuje plnou podporu pro správu konference. Více informací o systému naleznete na [5]. Tato aplikace opět využívá toho, že se veškerá agenda, která souvisí se správou konference, nachází a spravuje na serveru výrobce tohoto systému. Jde tedy opět o centralizovaný software, který se neinstaluje na server uživatele. Jedná se o aplikaci využívající technologií ASP. Jednotliví uživatelé se zaregistrují do systému a zasílají články k dané konferenci, kterou vytvořil správce konference. Tento systém nepřináší nic nového, co by již nebylo zmíněno u systémů, které jsme si popsali dříve a i přes to není tento systém zdarma. Zdarma je pouze možnost využít tento systém na 30 dní na ozkoušení nebo použít online demo verze projektu. Cena systému se také jistě odvíjí od faktu, že technologie ASP je od firmy Microsoft. Nicméně klasické ASP již není tolik využívané a cena 499\$ je velice vysoká, když bereme v úvahu, že existují bezplatné alternativy, které i v profesionální verzi stojí polovinu ceny v porovnání s tímto systémem.

2.4 Porovnání současných systémů

Tabulka 1 uvádí všechny zmíněné systémy a ke každému z nich je uveden seznam základní funkcionality a informace, jestli daný systém tuto funkcionalitu podporuje. Symbol „?“ znamená, že danou informaci nebylo možné ověřit.

Funkce	Název systému				
	Open Conference Systems	EasyChair	OpenConf		Conference Review System
			Community	Proffesional	
<i>Cena</i>	FREE	FREE	FREE	250\$	499\$
<i>Online Demo</i>	ANO	NE	NE	ANO	ANO
<i>Podpora</i>	ANO	ANO	NE	ANO	ANO
<i>Lokalizace</i>	V přípravě	NE	NE	NE	NE
<i>Vlastní instalace na server</i>	ANO	NE	ANO	ANO	NE
<i>Hosting u dané společnosti</i>	ANO	ANO	NE	ANO	ANO
<i>Podpora emailových zpráv</i>	ANO	ANO	ANO	ANO	ANO
<i>Automatické přidělení recenzentů</i>	NE	ANO	ANO	ANO	NE
<i>Programovací jazyk</i>	PHP	CGI	PHP	PHP	ASP
<i>Databáze</i>	MySQL	?	MySQL	MySQL	MSSQL

Tabulka 1: Srovnání současných konferenčních systémů

Z tabulky je zřejmé, že jako asi nejpříjemnější systém se jeví OCS, který je zcela zdarma, lze si jej vyzkoušet, má silnou podporu, chystá se česká lokalizace a systém lze nainstalovat na vlastní server nebo lze využít služby u společnosti vyvíjející tento systém, které však není zdarma a je tedy potřeba za ni zaplatit. Neplatí se však za systém jako takový ale pouze za pronájem prostoru na cizím serveru. Jediným kritériem, které tento systém nepodporuje je automatické přidělení recenzentů k článkům. Tuto činnost ale stejně většinou provádí organizátor konference sám.

3 SCRUM

Scrum je relativně nová metoda agilního programování, která menším týmům dovoluje v krátkých iteracích vyvíjet produkty s dobře odhadnutelnou cenou a podle zákaznických požadavků. Další informace naleznete na [7]. Této metodice bude věnována celá kapitola z důvodu použití této metodologie v případové studii v další kapitole této práce. V metodice Scrum se nachází hodně odborných názvů, a jelikož neexistují jejich doslovné překlady v češtině, budou zde používány jejich pravé anglické názvy. Základní pojmy budou vysvětleny v následující podkapitole. Metodika byla představena Kenem Schwabem a Mikem Beedlem v roce 2002. Tato metodika se přirovnává ke hře rugby, kde výraz scrum (mlýn) znamená skrumáž hráčů, kteří se snaží společnou týmovou silou dostat míč za čáru. V našem případě tedy společnou silou v týmu dotáhnout projekt do zdárného konce. Stejně tak jako ve sportu, se zaměřuje tato metodika na nečekané události a snaží se tedy rychle reagovat na případné změny nebo nenadálé situace. V této metodice se například od Extrémního Programování nebo Test-Driven Development zabýváme spíše organizačními a manažerskými aspekty vývoje software, než psaním samotného programového kódu. Při Scrumu je užitečná (není to bezpodmínečně nutné, ale tento postup patří k oblíbeným) přítomnost zákazníka při vývoji. Další informace o této problematice lze nalézt na [8]. Pak se zákazník účastní diskuzí o vývoji, odpovídá průběžně na otázky a pomáhá tak vyvíjet skutečně užitečný produkt vyrobený na míru zákazníkovi.

3.1 Pojmy

V této metodice se dostáváme do konfrontace se spousty nových termínů a je proto potřeba si nejprve tyto pojmy představit a vysvětlit. Zde se tedy nachází seznam stěžejních pojmů.

3.1.1 Scrum

Jedná se o kratší druh iterace, která trvá pouze jeden den. Na začátku každého Scrumu je uspořádán tzv. Scrum Meeting, který bude popsán později. Pomocí jednodenních iterací jsou vývojový tým i ostatní zúčastnění role neustále informovány jak probíhá vývoj a je tedy velice rychle možné řešit případné problémy.

3.1.2 Sprint

Je delší iterace, která trvá přibližně od 14 dní do jednoho měsíce a počet jednotlivých sprintů závisí na velikosti projektu. Doporučuje se mezi třemi až osmi sprinty. Na začátku každého sprintu, je plánovací schůzka, na které se všichni zúčastnění dohodnou, co se v rámci tohoto sprintu udělá a vytvoří se tzv. Sprint Backlog. Po dokončení sprintu je opět schůzka, na které si všichni popovídají o tom, co vše se stihlo, co se nestihlo vypracovat, kde byly problémy a jestli se přišlo na nějaké nové požadavky. Více se lze dočíst na [9].

3.1.3 Plánovací poker

Je taková zajímavost jak odhadovat pracnost položek v Product Backlogu. Jedná se o produkt Planning Poker, tedy pokerové karty, které obsahují čas úkolu. Autor si všiml, že klasické postupné dotazování jednotlivých členů týmu nejen že ovlivňuje ty ostatní, ale někteří nemusí

ani zapojit svůj mozek, pouze zopakují předchozího kolegu. Pomocí karet, kdy každý musí vybrat odhadovaný čas nezávisle na ostatních, nejen že donutí všechny popřemýšlet, ale neovlivňují se ani mezi sebou. Odhalí se tím pádem případné velké rozdíly v odhadech, o kterých se může dále diskutovat a odhalit skryté problémy.

Nachází se zde i speciální karty s nulou, což znamená buďto hotový nebo velmi jednoduchý úkol, a symbol kafička znamená jednoduše „*jsem unavený, chce to pauzu*“. Tímto jednoduchým způsobem může i každodenní porada vývojářů být celkem zábavná a co víc – úspěšnější. Více na [7].

3.1.4 Release

Celý výsledný produkt lze rozdělit do několika částí neboli releasů. U menších projektů není toto rozdělení na releasy potřeba ale například u projektů, které trvají například rok a více, je dobré dodat zákazníkovi například po jednom čtvrtletí alespoň první verzi finálního produktu s již hotovými funkcemi.

3.2 Role

Ve Scrumu figurují dvě skupiny tzv. Pigs a Chickens. Toto rozdělení vyplývá z povídky o praseti a kuřeti. Rozdíl mezi prasetem a kuřetem na projektu, do kterého vkládá prase maso a kuře vejce je v tom, že prase je „*zapojeno*“, zatímco kuře se problémem „*pouze týká*“. Proto je i ve Scrumu zvykem rozdělovat zúčastněné osoby do těchto skupin. Následují „*Pigs*“ role.

3.2.1 Vlastník produktu (Product owner)

Osoba, která zodpovídá za priority, za to, co se bude v příštím sprintu implementovat a určuje implementační detaily. Tato osoba je na rozhraní mezi zákazníkem a dodavatelem. Hlavní odpovědností je Product Backlog a také to, že aktuální backlog bude pro všechny zúčastněné dostupný.

3.2.2 Scrum master

Není to teamleader v klasickém slova smyslu, ale pracuje jako mezičlánek mezi týmem a jakýmkoli rušivým elementem zvenku. Řídí vývojáře, ale zároveň odstraňuje veškeré překážky, které brání programátorům ve vývoji. Mezi to patří například zajištění dostatečně silných strojů pro vývoj nebo také platné licence k placenému software. Je odpovědný také za to, že proces vývoje probíhá v souladu s pravidly a praktikami metodiky Scrum. Není přípustné aby Scrum Master byl v jedné chvíli zároveň programátor, jelikož by nebyl schopen zajistit programátorům odstraňování překážek. Ve většině případů se v této roli nachází Project Manager. Další informace naleznete na [10].

3.2.3 Scrum tým

V této skupině se nachází vývojáři, kteří pracují na vyhotovení jednotlivých položek z Product Backlogu. Členové týmu ovšem musí dělat i další činnosti jako je výběr jednotlivých položek z Product Backlogu a jejich následné zapsání do Sprint Backlogu. Také musí rozhodnout o pracnosti dané položky a učinit odhad pracnosti aby mohl být sestaven konečný plán sprintu.

3.2.4 Manažeři

Jsou již v roli Chickens. Osoby, které pomáhají nastavit prostředí. Mají na starosti konečná rozhodnutí o projektu a uzavírají smlouvy se zákazníkem. Podílejí se také na sestavování požadavků a cílů projektu i jednotlivých fází projektu.

3.2.5 Stakeholderi

Tyto osoby jsou také Chickens. Mezi tyto osoby patří uživatelé, zákazník ale třeba i testéři. Podílejí se hlavně na sestavení Product Backlogu, jelikož právě oni nakonec budou s výsledným produktem pracovat a proto právě oni by měli mít nejvíce připomínek a nápadů.

3.3 Artefakty

3.3.1 Product Backlog

Nejdůležitější dokument celé této metodiky. Do tohoto dokumentu se zapisují jednotlivé položky, které by měl výsledný produkt obsahovat. V dokumentu se tedy nachází slovní popis požadavků a také prioritizace požadavků, kterou určuje zákazník. Product Backlog by měl být co nejvíce přehledný a přístupný pro všechny role. Do Product Backlogu se průběžně položky přidávají, modifikují a odstraňují. O tento dokument se stará Product Owner a vývojový tým pak z něj vybírá jednotlivé položky, které budou ve Sprint Backlogu.

3.3.2 Sprint Backlog

Každý Sprint Backlog si bere určitý počet funkcí definovaných v Product Backlogu, které rozděluje na konkrétní úkoly pro konkrétní členy týmu. Právě členové týmu samotní pak odhadují, jak dlouho jim splnění přiděleného úkolu zabere. Výsledný Sprint Backlog nám tak dává informaci o tom, jaké funkce budou do nového produktu po skončení Sprintu přidány, kdo bude mít na starosti jakou část projektu, a hlavně, kolik práce daný Sprint zabere.

3.3.3 Burn-down chart

Sprint přichází s tzv. Burn-down grafem, což je zcela nový přístup ke grafickému znázornění časové náročnosti projektu. Ten na ose X zobrazuje celkový počet dní zbývajících do konce Sprintu a na ose Y ukazuje celkový počet práce zbývajících do splnění všech úkolů v daném Sprintu. Úsečka spojující body se souřadnicemi [Celkem zbývajících hodin, 0 dnů] a [0 zbývajících hodin, celkový počet dnů] se nazývá Ideal burn-down a ukazuje hypotetický ideální průběh prací pro splnění příslušného termínu. Kromě ní je na grafu zobrazena také křivka skutečného počtu zbývajících hodin, která se odvozuje od celkového součtu zbývajících hodin pro splnění jednotlivých úkolů v daný den. Tento počet se přitom může i zvyšovat, a to tehdy, kdy došlo například ke špatnému odhadu pracnosti některého z úkolů. Pokud se tato křivka nachází pod křivkou Ideal Burn-down, je projekt v předstihu, pokud nad ní, má projekt zpoždění. Výhodou tohoto grafického znázornění je přitom fakt, že každý den lze křivku skutečného počtu zbývajících hodin proložit regresní křivkou, která nám ukáže, jak si na tom aktuálně stojíme v porovnání s vytyčeným datem dokončení. V případě, kdy tak Burn-down chart indikuje několik dní po sobě zpoždění, má projektový manažer možnost analyzovat

nejkritičtější (tj. nejdéle trvající) úkoly a přidělit k nim další členy týmu, aby se zpoždění začalo včas dohánět. Více o této problematice naleznete v [11].

3.4 Scrum schůzky

Důležitou součástí agilního projektového managementu metodou Scrum jsou také tzv. Scrum meetingy. Ty mají především tři úkoly – včas rozpoznat začínající problémy, ověřit si pochopení přidělených úkolů u jednotlivých členů týmu a v neposlední řadě také zlepšit osobní nasazení jednotlivých členů týmu v jim přidělených úkolech.

3.4.1 Daily Scrum Meeting

Daily Scrum, který probíhá na začátku každého dne a trvá 15 až 30 minut, během kterých každý člen týmu stručně představí, co dělal včera, co bude dělat dnes, a zda se nevyskytly nějaké problémy, které by mohly ohrozit splnění jemu přiděleného úkolu. U náročnějších projektů s větším množstvím týmů pak následuje také Scrum of Scrums, kde se jednotlivé týmy zaměřující na to samé, ale tentokrát jen u úkolů, které se týkají (potažmo ovlivňují činnost) více týmů zároveň.

3.4.2 Sprint Planning Meeting

Zde dochází k již zmíněnému rozpadu Product Backlogu na jednotlivé úkoly pro daný Sprint. Rozdělování úkolů mezi členy týmu přitom může probíhat jak direktivně, tak demokraticky. Během tohoto prvního meetingu, který by měl trvat maximálně osm hodin, se také (po rozdělení úkolů) dospěje k celkové časové náročnosti daného Sprintu, na základě čehož může projektový manažer provést také relativně přesnou cenovou kalkulaci. V rámci úplně prvního Sprint Planning Meetingu také proběhne za účasti Product Ownera rozpad Product Backlogu na jednotlivé Release Backlog.

3.4.3 Sprint Review Meeting

Sprint Review Meeting je setkání, kde se shrne vše, co v daném Sprintu bylo a nebylo dokončeno a proběhne prezentace hotového funkčního celku zákazníkovi/zadavateli. Tento meeting by měl trvat maximálně 4 hodiny.

3.4.4 Sprint Retrospective

Je schůzka, které se zúčastní všichni členové týmu včetně projektového manažera. Toto setkání by přitom mělo přinést odpovědi na otázky, jak probíhal právě dokončený Sprint, kdo při něm pracoval dobře a co by šlo v příštím Sprintu zlepšit. Délka této schůzky by neměla přesáhnout tři hodiny.

3.5 Fáze vývoje

3.5.1 Předehra

V této úvodní fázi celého procesu se jako první začne vytvářet Product Backlog. Na tvorbě tohoto dokumentu se podílí zejména zákazník, který alespoň zhruba vypíše požadované funkce systému.

3.5.2 Hra

Toto je hlavní část celé metodiky Scrum a právě zde je produkt vytvářen. Tato fáze se dělí do tzv. sprintů, kterých se doporučuje obvykle 3-8 a každý z nich trvá 14 dní až jeden měsíc. V každém ze sprintů se pak implementují funkce vypsání ve Sprint Backlogu, který je vlastně částí Product Backlogu. Po každé iteraci je pak možné prezentovat zákazníkovi výsledek sprintu. Veškerý vývoj je tedy soustředěn v této fázi.

3.5.3 Dohra

Do této poslední fáze se dostáváme v momentě, kdy jsou vypracovány všechny položky z Product Backlogu a zákazník již nemá žádné další požadavky. Proběhne poslední release produktu a ten se předá zákazníkovi a otestuje. Až v této fázi je také tvořena dokumentace, což je jistě rozdíl oproti jiným metodikám jelikož ve Scrum není na dokumentaci kladen takový důraz jako v metodikách jiných. Více naleznete na [12].

4 Systém RTT

Tato kapitola se zaměřuje na vyvíjený systém a mapuje jeho požadavky na funkčnost. V závěru budou představeny reálné výstupy systému a statistiky.

4.1 Ukázkový projekt

Cílem tohoto ilustračního projektu je realizace konferenčního systému, jejichž specifika byla popsána v druhé kapitole. Jde o systém, který je vytvořen na základě reálného zadání.

Zadavatelem je Katedra telekomunikační techniky na Fakultě elektrotechniky a informatiky na Vysoké škole báňské - Technické univerzitě (dále již jen jako zadavatel). Katedra je jednou z 3 kateder v České republice poskytující vysokoškolské vzdělání v oboru telekomunikační techniky. Jelikož se zadavatel zabývá výzkumnými činnostmi, je zde potřeba zavést systém, který pomůže se správou a evidencí chystaných odborných konferencí na půdě zadavatele.

Vzhledem k tomu, že cílem této části práce není podrobná analýza projektu, ale přiblížení a popis výsledného systému, jsou tyto analýzy pokrývající celý vývojový proces projektu obsahem přílohy.

4.2 Specifikace požadavků

4.2.1 Funkční požadavky

4.2.1.1 Proč nový systém?

Přehled známých konferenčních systémů, které byly uvedeny v druhé kapitole, ukazuje, že na trhu je dostatek konferenčních systémů, ze kterých si potenciální zákazník může vybrat, i když tento výběr nemusí být zrovna jednoduchý. Nastává otázka, proč pro výše zmíněný vyvíjený systém nebylo použito stávající řešení. Ať už jako hotový produkt, nebo na míru pro naše potřeby upravený systém, pokud to licence konkrétního systému dovoluje. Výhody pro nasazení již hotového a odladěného řešení jsou poměrně jasné, níže jsou uvedeny ty nejdůležitější.

- Stávající řešení jsou skutečně používána a odladěná
- Existuje podpora a dokumentace
- Nástroje jsou komplexní s množstvím funkcionality
- Nižší nebo dokonce nulová cena
- Časová úspora a možnost okamžitého nasazení

I přes tyto jasné fakty jsme se rozhodli vytvořit úplně nový systém. Ani jeden z konferenčních systému nesplňoval všechny naše požadavky. Doposud na katedře neexistuje žádný konferenčně informační systém. Systémy, které existují a jsou dostupné na internetu, nejsou podle představ zadavatele. Navíc zadavatel chce systém svůj, který může dále vyvíjet a upravovat. Evidence by jinak musela být vedena ručně a to není z důvodu velikosti dat vhodné a nabízí se automatizace v podobě vytvoření informačního systému. Většina konferenčních systémů je složitá a po

vstupu do systému uživatel často vidí mnoho voleb a přeplněných menu. Při přidávání nebo recenzování článku musí uživatel vyplnit velké množství formulářů a dalších údajů. I když jsou dostupné spousty programů, ne všechny jsou použitelné a hodně organizací vyvíjí a spravuje svůj vlastní systém. Cílem je vytvoření jednoduchého a použitelného systému, který dělá přesně to, co potřebujeme a nic navíc. Další důvody, které nás vedly k vytvoření nového konferenčního systému, jsou následující.

- Lokalizace – drtivá většina stávajících řešení nejsou lokalizovaná do českého jazyka. Většina systémů komunikuje s uživatelem v anglickém jazyce. Lokalizace byla jedním z požadavků na systém (aspoň v počátku, později při vývoji byla nakonec lokalizace odstraněna). Většina systémů nemá jazykové texty uložené v samostatném soboru. Překlad systému byl pak velmi složitý a časově náročný, jelikož by se všechny textové řetězce musely hledat ve zdrojových kódech.
- Požadavkem je také to, aby mohla být aplikace nainstalovaná na serveru zákazníka. Nepředpokládá se využití placené služby některé ze společností poskytující webový prostor.
- Jedním z důležitých a ve své podstatě paradoxním nedostatkem, je komplexnost nabízených konferenčních systémů. Podporují veliké množství úloh a jsou schopné spravovat velké světové konference. Systémy většinou nejsou modifikovatelné takovým způsobem, že bychom si mohli nastavit požadované vlastnosti.
- Zadavatel má jasné představy o požadované funkcionalitě konferenčního systému, což umožňuje vytvořit systém přímo na míru jeho zadání.

Více informací lze najít v [1].

4.2.1.2 K čemu má systém sloužit?

Informační systém má sloužit v administraci chystaných konferencí, které se uskuteční v rámci školy. Je tedy potřeba vést potřebné evidence: uživatelů, článků, recenzí, termínů, výborů konferencí, programů, poplatků, ubytování, dopravy a dalších položek. Zhotovitel navrhne jak systém, tak i grafický vzhled systému a zvolí si programovací jazyk a technologie, které k jeho tvorbě použije. Celý systém bude zajišťovat kompletní služby od registrací na konferenci, zasílání abstraktů, článků, recenzí až po možnosti ubytování v hotelu. Celý systém poté nasadí na vybraný školní server a uvede do provozu. Registrací se daný účastník bude moci přihlásit do konference, kde bude možnost zasílání příspěvků, vybrání si dnů, kdy se zúčastní konference, platba za účast, případně pokud bude vybrán, bude provádět recenze dle mezinárodních standardů jako je IEEE, SPIE, OSA a abstraktů na konferenci. Účastník konference bude pak moci v systému editovat a nahrávat příspěvky. Hlavní pořadatel konference poté přidělí k jednotlivým příspěvkům recenzenty, kterým dojde automaticky na jejich zaregistrovaný emailový kontakt zpráva o žádosti provedení recenze. Daný recenzent pak provede recenzování daného článku přímo v systému. Dále pro přehledný soupis a počet lidí nutných k dopravení se na místo konání, bude vytvořeno místo, kde účastníci definují způsob dopravy a pokud bude třeba, budou moci využít autobusu školy, a bude se automaticky upravovat počet volných míst v

autobusu a rezervace míst podle přihlášení se způsobem „*kdo dříve se přihlásí, má přednost*“. Rovněž bude v systému konferencí záložka pro ekonomickou část, kde účastník dostane informaci, kolik jej bude stát konference, za zaslání článku, přihlášení do konference apod. To znamená vložné za příspěvek, ubytování, dopravu a další dodatečné služby.

4.2.1.3 Kdo bude se systémem pracovat?

S informačním systémem budou pracovat běžně uživatelé, kteří se na konferenci přihlásí a autoři, kteří pošlou k prezentování svůj příspěvek. Dále zde budou vystupovat recenzenti, kteří budou administrátorem nebo správcem recenzí vybráni k tomu, aby ohodnotili zasláný příspěvek. K administraci a konfiguraci konference se zde bude vyskytovat administrátor.

4.2.2 Vstupy do systému

- **Příspěvky**

Uživatel spravuje své příspěvky a má možnost s nimi dále pracovat. Správce může spravovat příspěvky všech uživatelů a nastavovat jim stav příspěvku. V závislosti na vyhodnocení systému recenzí nastavuje systém stav danému příspěvku.

- **Uživatelé**

Spravovat uživatele může pouze administrátor. Může vytvářet nové, odstraňovat, upravovat a nastavit role v jakých v systému vystupují. Při registraci uživatel vloží informace o své osobě. Systém mu automaticky nastaví roli obyčejného uživatele. Pomocí rolí dále evidujeme, zda je uživatel správcem či ne.

- **Konference**

Budeme evidovat vytvořené konference a u nich hlavně jestli se jedná o aktivní konferenci, po jejímž aktivování se přizpůsobí celkový vzhled a nastavení systému.

- **Témata konference**

Je potřeba evidovat témata každé konference. Podle témat konference se pak budou řadit články. Autor článku při zasílání článku musí vždy vyplnit, do jakého tématu chce článek zařadit.

- **Webové stránky konference**

Každá konference má k dispozici své vlastní webové stránky. V těchto stránkách jsou uloženy všechny nezbytné informace o konferenci.

- **Termíny konference**

Administrátor potřebuje evidovat termíny každé konference a nastavit tyto termíny tak aby systém mohl například automaticky zastavit přijímání článků nebo registrací do konferencí.

- **Přihlášení ke konferencím**

Pro správce systému je nutné, aby měl seznam všech přihlášení ke konané konferenci. Uživatel se automaticky přihlašuje ke konferenci při registraci ke konferenci, která je právě nastavena jako aktivní.

- **Možnosti dopravy na konferenci**

Na každou konferenci budou moci být různé možnosti dopravy. Pouze jedna možnost však bude moci být nastavena k odpočítávání volných míst například v autobuse. Tato sekce je přístupná pouze administrátorovi systému.

- **Recenze článků**

Jelikož recenze jsou jedním z hlavních požadavků na systém, je zřejmé, že budou evidovány. Recenze přiděluje správce tím, že vybírá jednotlivé recenzenty a přiděluje jim články.

- **Sborníky článků**

Každá konference má sborník článků, které se nakonec presentovaly na reálné konferenci. Budeme tedy evidovat články, které byly přijaty do sborníků, a sborník bude dále vystaven na stránkách systému.

- **Otázky v recenzích**

Jelikož pro každou konferenci bude potřeba mít různé otázky, je potřeba evidovat i otázky nacházející se v recenzích. Tyto otázky tedy dále tvoří recenzní formulář.

- **Účtované položky**

Uživatel bude potřebovat podrobný přehled o účtovaných položkách, které bude muset později zaplatit. Správce systému bude zase potřebovat podrobný seznam všech účtovaných položek jednotlivým uživatelům.

- **Peněžní účty**

Tyto účty se skládají z jednotlivých účtovaných položek a budeme tedy evidovat celkové náklady každého uživatele v systému.

- **Programy konference**

Program se skládá z jednotlivých událostí a proto program je pouze seznamem událostí k dané konferenci, který ovšem bude potřeba dále vystavit na webových stránkách konference.

- **Události na konferenci**

Konference bude potřebovat seznam všech událostí, které se budou v rámci reálné konference provádět. Jednotlivým událostem bude možno přidělit prioritu.

- **Automatické e-maily**

Každá konference bude zasílat množství emailových zpráv autorům o jejich člancích. Proto bude potřeba evidence těchto zpráv a jejich modifikace.

- **Nastavení aplikace**

Zbývá nastavení aplikace, která mohou být různá pro dané konference. Např. IP adresa SMTP serveru.

4.2.3 Výstupy ze systému

- **Příspěvky**

Jednotlivý uživatelé budou mít seznam všech svých příspěvků a budou je moci modifikovat. Správci budou mít podrobné seznamy všech příspěvků a budou je moci spravovat podle potřeb.

- **Uživatelé**

Každý uživatel bude moci modifikovat a zobrazovat si svůj profil v systému. Správci budou moci zobrazovat jednotlivé profily uživatelů a také zobrazovat sestavy podle různých kritérií jako např. kdo se bude dopravovat na reálnou konferenci autobusem, kdo se na konferenci přihlásil apod.

- **Konference**

Seznam všech konferencí uložených v systému.

- **Témata konference**

Seznam všech témat vyskytujících se v konferencích a možnost jejich modifikace.

- **Webové stránky konference**

V jednotlivých stránkách je možnost editace těchto stránek pomocí speciálního editoru, který umožní editaci přímo v aplikaci. Výsledný dokument zobrazí prohlížeč jako webovou stránku. Správce tak může spravovat stránky konference kdekoliv.

- **Termíny konference**

List všech termínů k jednotlivým konferencím. Tato sekce je pouze k službám správci.

- **Přihlášení ke konferencím**

Výpis veškerých přihlášení k daným konferencím. Tyto sestavy slouží pouze pro evidenci správci systému.

- **Možnosti dopravy na konferenci**

Seznam možností jak se dopravit na reálnou konferenci a nastavení jediné možnosti, která bude odpočítávat volná místa při registraci do konference pro účely zajištění dopravy přímo katedrou telekomunikací.

- **Recenze článků**

Správce systému a recenzí potřebuje seznam všech recenzí, které přidělil jednotlivým uživatelům. Zde může s recenzemi pracovat a zobrazovat jejich výsledky.

- **Sborníky článků**

Výsledný seznam všech článků, které byly na konferenci prezentovány a přijaty k oficiálnímu publikování. Tento obsah bude přístupný pro všechny návštěvníky systému a sborníky budou uvedeny na webových stránkách.

- **Otázky v recenzích**

Seznam všech otázek vyskytujících se v recenzích pro danou konferenci a jejich následná modifikace. Tato možnost je pouze pro správce systému.

- **Účtované položky**

Seznam položek, které se vyskytují v peněžním účtu některého z uživatelů. Administrátor bude moci evidovat tyto položky a případně modifikovat jejich hodnoty.

- **Peněžní účty**

Seznam všech účtů uživatelů a informace o jejich stavu platby. Velice důležitá sekce přístupná pouze pro správce. Uživatel bude moci zobrazit pouze svůj účet.

- **Programy konference**

Sestava programu konference, kterou vytvoří správce. Tato sestava bude dále zobrazena na webových stránkách systému pro přehled toho, jak bude konference probíhat.

- **Události na konferenci**

Výpis všech událostí konaných v rámci konference. Správce bude moci nastavovat priority jednotlivých událostí.

- **Automatické e-maily**

Výčet všech automatických e-mailů a možnost jejich editace správcem aplikace.

- **Nastavení aplikace**

Seznam zbylých nastavení systému. Sekce přístupná pouze pro správce.

4.2.4 Vybrané funkce systému

- **Vytvoření uživatele**

Uživatel je vytvořen buď administrátorem, nebo vlastní registrací své osoby. Uživatel vyplní povinné položky a je mu umožněno přihlásit se do systému.

- **Zaslání článku**

Zaregistrovaný uživatel má možnost zaslat svůj článek do databáze, ale musí vyplnit povinné údaje o příspěvku.

- **Recenzování článku**

Správce přidělí uživateli recenzi k článku a ten po přihlášení do systému vyplní recenzní formulář.

- **Přihlášení ke konferenci**

Uživatel má možnost přihlásit se k aktuální konferenci.

- **Zasílání informativních e-mailů**

Systém automaticky rozesílá emailové zprávy svým uživatelům na jejich zaregistrované e-maily například v případě, kdy se zaregistruje, zašle článek, článek uživatele proběhne recenzním cyklem apod.

- **Zobrazení poplatků za konferenci**

Uživatel má možnost si zobrazit podrobný přehled o jeho položkách, které bude muset v budoucnosti zaplatit.

- **Nastavení aplikace**

Správce systému nastaví různá nastavení systému pro jednodušší správu. Od IP adresy SMTP serveru až po počet míst v autobuse připraveném k dopravě na reálnou konferenci.

4.2.5 Nefunkční požadavky

Od výsledného konferenčního systému bude požadováno dodržení nejnovějších standardů a předpisů pro tvorbu informačních systémů. K systému bude v jednu chvíli přes internet přistupovat velké množství uživatelů. Je tudíž nutná vhodná implementace klient – server, která nám zajistí rychlou odezvu na požadavky uživatelů. Systém bude přístupný přes všechny standardní webové prohlížeče. Celý informační systém bude vyvíjen v .NET Frameworku – programovacím jazykem C# technologií ASP MVC s pomocí MS SQL Serveru.

4.2.6 Diagram případů užití

Modelování případů užití je jednou z forem inženýrství požadavků. Modelování případů užití je jiným, doplňkovým způsobem získávání a dokumentování požadavků, více o této problematice v [13]. Modelování případů užití se skládá z následujících aktivit:

- Nalezení hranic systému
- Vyhledání aktérů
- Nalezení případů užití
- Uvedený postup je třeba opakovat, dokud nejsou ustáleny výše zmíněné aktivity

Obrázek 1 znázorňuje diagram případů užití vyvíjeného systému.



Obrázek 1: Use Case diagram

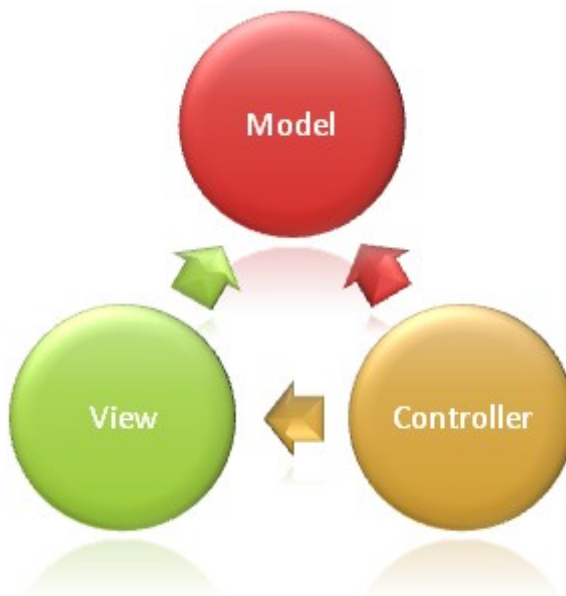
4.3 ASP.NET MVC Framework

Častým problémem při vývoji webových aplikací je nedostatečné oddělení aplikační logiky od prezentační vrstvy aplikace, a proto zde bude uveden menší popis použité technologie při vývoji tohoto projektu. V klasických Windows Forms se striktně nevyžaduje oddělení těchto dvou vrstev. U menších webových aplikací tato skutečnost nemusí vadit avšak v u větších aplikací

jako je například výše zmíněný systém je potřeba zavést jakýsi řád. O zavedení tohoto řádu se pokouší architektonický vzor MVC (Model, View, Controller). Proto byla vybrána pro implementaci systému tato technologie, která se v kombinaci s ASP.NET nazývá ASP.NET MVC, což je poměrně mladá technologie se spoustou zajímavých schopností a rozhodně slibnou budoucností. ASP.NET MVC slouží jako alternativa k dobře známým ASP.NET WebForms (v žádném případě se je nesnaží nahradit), stejně jako klasické WebForms spolupracuje tato technologie se stávajícími vlastnostmi ASP.NET. Pro tuto technologii také hovoří tyto výhody.

- Usnadňuje správu aplikace rozdělením do tří komponent.
- Nepoužívá View State, ani serverové ovládací prvky. Díky tomu je MVC framework ideální pro ty, kteří chtějí mít kompletní moc nad chováním aplikace.
- Používá tzv. „*Front Controller*“, který zpracovává požadavky od aplikace pomocí jednoho dalšího Controlleru.

Jak spolu komunikují jednotlivé vrstvy tohoto architektonického vzoru, znázorňuje Obrázek 2. Více informací lze dohledat na [6].



Obrázek 2: Architektonický vzor MVC

4.4 Nasazení do provozu

Pro nasazení ASP.NET MVC aplikace na server zákazníka je zapotřebí provést určitá nastavení, která jsou vypsána níže.

- Operační systém
- Internetová Informační služba

- Konfigurační soubor aplikace
- MSSQL Server

Po nastavení těchto položek seznamu, je systém nainstalován na server a tím připraven na ostrý provoz a předán zákazníkovi.

4.5 Ukázky aplikace

Pro lepší představu jak výsledný konferenční systém vypadá, jsou zde prezentovány obrázky vyvíjené aplikace. Jako nejdůležitější část celého systému jsou pro zákazníka výsledné výstupní sestavy, které mu dodávají celkový pohled. Obrázek 3 znázorňuje výstupní sestavu, kde jsou v podrobném seznamu vypsáni všichni uživatelé dané konference.

Obrázek 4 představuje výstupní sestavu všech článků. Administrátor má možnost zobrazovat seznam podle jednotlivých témat konference, nebo podle jednotlivých uživatelů a jejich příspěvků. Na obrázku si lze také všimnout vzhledové koncepce. V levé části je menu, pod kterým se nachází prostor pro logo katedry a exkluzivní partnery. V pravé části jsou loga zbylých partnerů a sponzorů konané konference. Uprostřed stránky je zobrazen hlavní obsah. V horní části je prostor pro název aktuální konference a tlačítko pro přihlášení / odhlášení.

RTT 2010

Logout

User administration

Number	Login - Email / User / Articles	Attendance / Travel	Organization	Actions
1	admin - admin@admin.com Admin Admin	I am not going to conference	VŠB	✖ ✖
2	ales.kumpera@seznam.cz - ales.kumpera@seznam.cz Ing. Aleš Kumpera, Ph.D.	I am not going to conference	Tempest Czech	✖ ✖
3	baronak - baronak@ktl.elf.stuba.sk prof. Ing. Ivan Baroňák, Ph.D.	By car	FEI STU Bratislava	✖ ✖
4	balman - matus.pleva@tuke.sk Ing. Matus Pleva, Ph.D.	I am not going to conference	KEMT FEI TU Kosice	✖ ✖
5	bencko - bencko@fel.uniza.sk Ing. Miroslav Benčo, Ph.D.	By bus from Ostrava	ŽU v Žilne	✖ ✖
6	bestar1 - bestar1@fel.cvut.cz Ing Robert Bestak, PhD	I am not going to conference	CTU in Prague	✖ ✖
7	bezpa - bezpalec@fel.cvut.cz Pavel Bezpalec	By car	FEE CTU in Prague	✖ ✖
8	bohac - bohac@fel.cvut.cz Ing. Leos Bohac, Ph.D.	By train to Zábřeh na Moravě - Šumperk - Velké Losiny	Czech technical university in Prague	✖ ✖
9	BoonRuk ChiPop - boonruk2000@yahoo.com Mr. Boonruk Chippop	I am not going to conference	KMUTT	✖ ✖
10	borowik - borowik@taty.net.pl Doc. Ing. Bohdan Borowik, Ph.D.	By car	University of Bielsko-Biala	✖ ✖

Supported by:

Partners:

T-Mobile Czech Republic a.s.

Sponsors:

PROFiber NETWORKING
PROFiber Networking CZ s.r.o.

2N TELECOMMUNICATIONS
2N TELEKOMUNIKACE a.s.

TR instruments
TR instruments, spol. s r.o.

SIEMENS
Siemens Enterprise Communications
Siemens Enterprise Communications, s.r.o.

ROHDE & SCHWARZ
ROHDE & SCHWARZ - Praha, s. r. o.

Obrázek 3: Administrace uživatelů v konferenčním systému RTT

RTT 2010

rtt.vsb.cz/Clanek/Index

Logout

Home

Call for papers

Time schedule

Venue and travel

Author information

Conference committees

Keynote speaker

Photogallery

Conference topics

Preliminary programme

Official language

Registration

Contact

My articles

My reviews

My fees

User profile

Change password

User administration

Article administration

Reviews administration

Time schedule administration

Pages administration

Travel administration

Topic administration

Questions administration

Fees administration

Article administration

Conference articles count: 52

Modern trends in telecommunications

Number	Username	Author	Organization	Title	Topic	Sent date	Status	Actions
1	Michal Jahelka	Ing. Michal Jahelka, Ph.D.	VSB-TU Ostrava	Communication system for intelligent houses	Modern trends in telecommunications	7/19/2010 11:42:42 AM	Accepted with reservations	
2	nov24	Radek Novák	VSB TU Ostrava	Radek Novák	Modern trends in telecommunications	7/16/2010 4:00:29 PM	Accepted with reservations	
3	novotnyv@fec.vutbr.cz	doc. Ing. Vít Novotný, Ph.D.	Bmo University of Technology	Hierarchical system of data acquisition with intermediate information processing	Modern trends in telecommunications	7/11/2010 8:21:02 AM	Accepted	
4	sentinel	Ing. Tomáš Kánčiz	FEI TUKE	Open Shortest Path First – a widespread routing protocol	Modern trends in telecommunications	7/9/2010 2:57:32 PM	Not accepted	
5	radovan_ridzon@tuke.sk	Ing. Radovan Ridzon, PhD.	KEMT FEI TU	SHA3 – New Hash Function	Modern trends in telecommunications	6/14/2010 4:54:39 PM	Accepted with reservations	
6	krišovsk	Doc. Ing. František Křížovský, CSc.	ČVUT v Praze FEL K13132	Service System M/G/1 (Distribution function of waiting times)	Modern trends in telecommunications	6/14/2010 10:43:02 AM	Accepted	

Multimedia systems

Number	Username	Author	Organization	Title	Topic	Sent date	Status	Actions
1	vyc022	Jiří Vychodil	VSB-TU	Approach to converting WAV container into PCAP	Multimedia systems	7/20/2010 6:48:12 AM	Accepted	

Supported by:

Partners:

T-Mobile

T-Mobile Czech Republic a.s.

Sponsors:

PROFiber

PROFiber Networking CZ s.r.o.

2N

TELECOMMUNICATIONS

2N TELEKOMUNIKACE a.s.

TR instruments

TR instruments, spol. s r.o.

SIEMENS

Siemens Enterprise Communications

Siemens Enterprise Communications, s.r.o.

ROHDE & SCHWARZ

ROHDE & SCHWARZ - Praha, s. r. o.

Obrázek 4: Správa příspěvků zaslaných do konferenčního systému RTT

Obrázek 5 znázorňuje, jak vypadá recenzní formulář. Recenzent zde nalezne veškeré potřebné informace ke článku a následně vyplní všechna požadovaná pole. Známkování probíhá zadáním známky od 1 do 10, kdy nejvyšší číslo má nejkvalitnější hodnotu. Po vyplnění známek, recenzent vybere výsledný verdikt o následném stavu článku.

RTT 2010

Logout

Article Review

1. Reviewer data

First name: Lukas
Last name: Vojtech
E-mail: vojted@fel.cvut.cz

2. Paper data

Title: Communication system for intelligent houses
Article: http://rtt.vsb.cz/Prispevky/Michal_Jahelka_Jahelka.pdf
http://rtt.vsb.cz/Prispevky/Finals/Michal_Jahelka_FINAL_RTT2010_Jahelka.pdf
Abstract: The Department of Telecommunications, at the Technical University of Ostrava is developing intelligent house. This hardware is to be equipped with audio communication that would enable it to communicate with any audio capable device throughout the house and outside of the house. This paper describes this system, digital message exchange and possibilities of extension in future.

If you cannot continue with the reviewing of this paper, please, send a message to the conference organizers by clicking [here](#)

3. Evaluation

Before recommending the acceptance (or non-acceptance) of the paper/abstract you reviewed, please use the maximum number of the following criteria to evaluate it on a scale from 1 to 10 (the number 1 (one) being associated to the lowest evaluation and the number 10 (ten) being associated to the highest one):

Originality: 5 Not known or experienced before. A technique or a method not used before. Has this or similar work been previously reported? Are the problems and/or approaches in the paper completely new?

Originality comment:
It is known problem - home communication system, but this paper shows "complete" system description. It could be interesting for other designers. Especially communication protocols.

Novelty: 6 According this criterion, is not necessary for the paper to develop new techniques, or to generate new knowledge, but it should, at least, apply, or combine, them in a fresh and novel way or shed some new light on their applicability in certain domain.

Novelty comment:
Described HW, SW and protocols combination is a good idea for a new home communication systems.

Supported by:

Partners:
T-Mobile
T-Mobile Czech Republic a.s.

Sponsors:
PROFiber
PROFiber Networking CZ s.r.o.
2N
TELECOMMUNICATIONS
2N TELEKOMUNIKACE a.s.
TR instruments
TR instruments, spol. s r.o.
SIEMENS
Siemens Enterprise Communications
Siemens Enterprise Communications, s.r.o.
ROHDE & SCHWARZ
ROHDE & SCHWARZ - Praha, s.r.o.

Obrázek 5: Recenzní formulář v konferenčním systému RTT

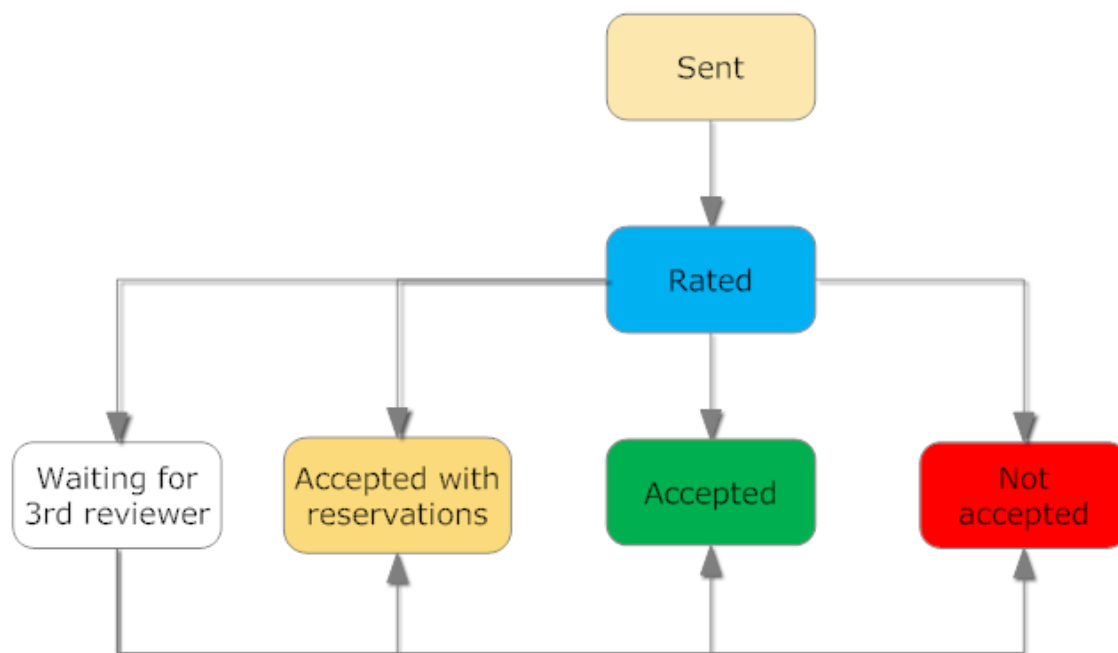
Tabulka 2 zobrazuje způsob, jakým jsou vyhodnocovány finální rozhodnutí o stavu článku.

1./2. Recenzent	Accepted	Accepted with reservations	Not accepted
Accepted	Accepted	Accepted with reservations	Waiting for 3rd reviewer
Accepted with reservations	Accepted with reservations	Accepted with reservations	Waiting for 3rd reviewer
Not accepted	Waiting for 3rd reviewer	Waiting for 3rd reviewer	Not accepted

Tabulka 2: Systém vyhodnocení stavu článku v konferenčním systému RTT

V prvním sloupci jsou možnosti, které mohou být vybrány prvním z recenzentů. V prvním řádku jsou ekvivalentně prvnímu sloupci vypsány možnosti, které mohou být vybrány druhým recenzentem. Systém pak po vybrání stavů obou recenzentů, nastaví danému článku stav, který se nachází v průniku prvního řádku a prvního sloupce. V případě, že systém nastaví článku stav „Waiting for 3rd reviewer“, vybere administrátor pro tento článek rozhodujícího třetího

recenzenta, který již definitivně rozhodne o stavu článku. Pro lepší znázornění je zde Obrázek 6, ve kterém je ještě uveden stavový diagram, na kterém je zachycen celý životní proces článku.



Obrázek 6: Stavový diagram životního cyklu článku v konferenčním systému RTT

Po zaslání článku do systému, je daný článek ve stavu „Sent“ (odeslaný). Ve chvíli, kdy je článku přidělena první recenze, se článek dostane do stavu „Rated“ (hodnocený). V tomto stavu je článek do té doby, dokud systém nerozhodne o dalším stavu na základě vyhodnocení, které bylo popsáno výše.

4.6 Statistiky systému

Celkově se zaregistrovalo 136 uživatelů, z celkového počtu zaregistrovaných uživatelů se fyzicky zúčastnilo konference 89 uživatelů, 29 uživatelů využilo možnosti dopravy na konferenci autobusem.

Celkově se zaslalo do systému 80 článků, které byly rozděleny do 8 konferenčních témat. Z celkového počtu článků bylo 52 článků konferenčních a 28 článků bylo workshopových. V konečné fázi nebyly přijaty pouze dva příspěvky, 21 konferenčních příspěvků bylo přijato do konference bez připomínek, a všech 28 workshopových článků bylo přijato.

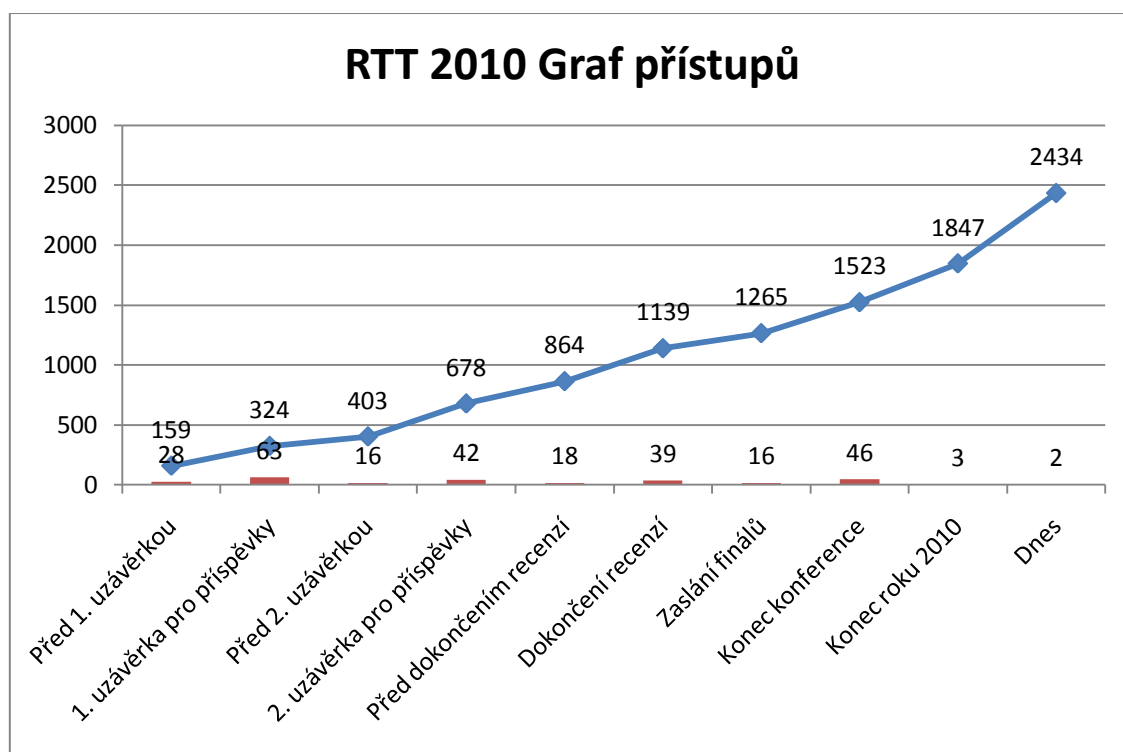
Celkem bylo provedeno 118 recenzí, z nichž 72 získalo status „Accepted“, 13x byl přidělen status „Not accepted“ a 33x byl zadán status „Accepted with reservations“.

Z uživatelů nebyl zaplacen dluh za konferenci pouze jedním uživatelem a zbylých 135 tedy zaplatilo svůj účet. Celkovou vybranou částku, celkovou nevybranou částku a celkovou vybranou částku za konferenci zde není možno z pochopitelných důvodů publikovat.

4.6.1 Google Analytics

Pro detailní statistiky přístupů bylo využito služby Google Analytics. Google Analytics je řešení webové analýzy pro podniky, které poskytuje dokonalý přehled o provozu na webových stránkách a o efektivitě marketingu. Díky výkonným, přizpůsobivým a snadno použitelným funkcím můžete nyní zobrazovat a analyzovat údaje o provozu zcela novým způsobem. Google Analytics umožňuje vytvářet lépe cílené reklamy, posilovat marketingové aktivity a vytvářet webové stránky s vyššími konverzními poměry. Více informací lze nalézt na [15].

Obrázek 7 znázorňuje, jaký byl průběh přístupů návštěvníků na nasazený systém. V grafu lze pozorovat, že těsně před ukončením první uzávěrky přijímání příspěvků systémem, bylo přístupů do systému průměrně 28, ale v den ukončení přijímání příspěvků byl tento počet dvojnásobný. I na tomto grafu je tedy vidět, že většina příspěvovatelů, nechala nahrávání článků do systému na poslední chvíli. Z tohoto důvodu byl termín prodloužen o necelý měsíc a před tímto prodlouženým termínem bylo přístupů kolem 16. V den ukončení druhé uzávěrky se ale opět počet přístupů více než zdvojnásobil. Stejně tak tomu bylo i v případě zadávání recenzí do systému. Před konečným termínem zaslání recenzí bylo přístupů průměrně 18 a v poslední den se toto číslo zdvojnásobilo. V době konání reálné konference bylo přístupů do systému průměrně do 50 a po ukončení konference již počet přístupů prudce klesl. Na konci roku 2010 to bylo kolem 3 přístupů denně a dnes již jen 2.



Obrázek 7: Graf přístupů RTT 2010

Z celkových 2434 přístupů bylo zobrazeno 6871 stránek konferenčního systému, což je průměrně 2,82 zobrazení stránek pro jeden přístup. Míra opuštění stránek se rovná 34%. Průměrná doba strávená na stránkách systému je 10:05 minut a podíl nových návštěv je 53,2%.

Webové stránky konané konference byly nejčastěji navštěvovány z internetového vyhledávače Google a to 53%. Zbýlých 47% byla přímá návštěvnost. Nejčastěji vyhledávanými výrazy do vyhledávače Google byly: „*rtt 2010*“ s 67% a „*rtt 2010 konference*“ s 15%.

Nejnavštěvovanější stránkou systému byla úvodní obrazovka, což je logické a dále jí následuje stránka „*Call for papers*“. Nejpoužívanější prohlížeč návštěvníků je Firefox dále pak Internet Explorer, Google Chrome a Opera.

Stránky byly nejvíce navštěvovány z Evropy konkrétně z České republiky 57%, Slovenské republiky 24%. Mezi dalšími zeměmi, které navštívili webové stránky konference je například Rusko, Maďarsko, Polsko, Německo. Jako nejzajímavější návštěvu lze vyzdvihnout přístup z Taiwanského města Nei-Hu.

5 Případová studie SCRUM

V této kapitole budou popsány kroky, které souvisí s vybranou metodikou při realizaci ukázkového projektu.

5.1 Proč Scrum?

Důvodů, proč byla vybranou agilní metodikou vývoje metodika Scrum je hned několik. Nyní bude uveden seznam těch nejhlavnějších.

- Výborně funguje v malých týmech
- Rychlá reakce na změny
- Žádné konkrétní implementační postupy
- Inkrementální vývoj
- Zpětná vazba
- Úzký kontakt se zákazníkem

Důvodů je samozřejmě více, ale ty už nějakým způsobem zapadají do výše jmenovaných. Ukázkový projekt byl vyvíjen v krátkém časovém rozmezí, a proto byla potřeba vybrat metodiku, která dodává zákazníkovi častý přehled o situaci, v jaké se projekt nachází. Zákazník sám měl potřebu podílet se na vývoji finálního produktu, což metodika Scrum přímo vyžaduje. Tato metodika neudává žádné konkrétní implementační postupy, což se ve většině případů bere spíše jako nevýhoda metodiky Scrum, avšak v tomto případě, kdy nebyl dostatek času, se rázem z nevýhody stává výhoda. Díky každodenní komunikaci se zákazníkem je zajištěna rychlá reakce na změny a vývojářům se tak dostává rychlé zpětné vazby.

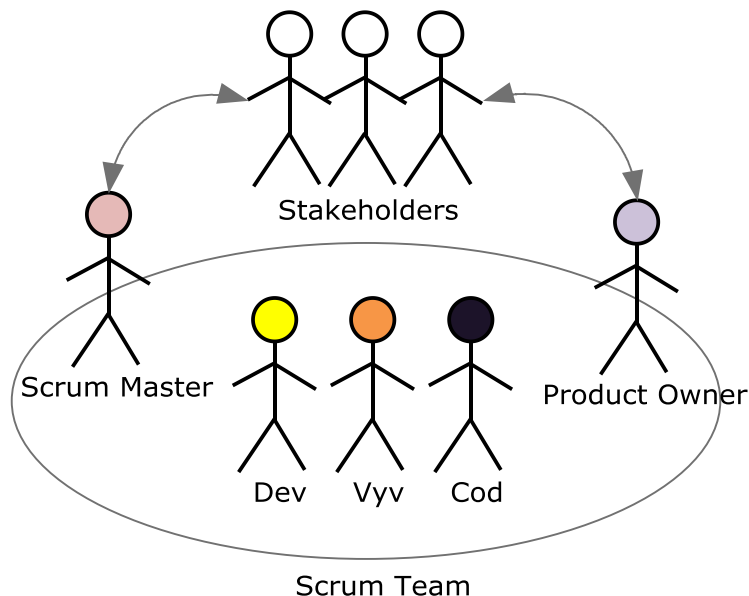
5.2 Použití Scrum

Nyní již přichází na řadu popis realizace ukázkového projektu. Nejprve bude představen vývojový tým, který celý produkt vyvinul, a následně budou popsány všechny tři fáze vývoje vyskytující se v této metodice.

5.2.1 Tým

Obrázek 8 ilustruje rozložení vývojového týmu. Tento tým se v našem případě skládá ze tří developerů označených Dev (Developer), Vyv (Vývojář), Cod (Coder), kteří se starají o vývoj jednotlivých položek z Product Backlogu, dále z jednoho vlastníka produktu a Scrum Mastera. Scrum Master a Product Owner se nachází na rozhraní mezi týmem a stakeholdery. Stakeholderi tak komunikují s těmito osobami, které následně předávají požadavky dále zbytku týmu. Product Owner, Scrum Master a zákazník sestavují společně Product Backlog na základě těchto požadavků. Scrum Master se mimo jiné stará o to, aby vývoj probíhal striktně podle praktik a zásad metodiky Scrum. Pokud se totiž rozhodnete pro tyto postupy, je potřeba je následovat úplně nebo vůbec jelikož napůl metodika nefunguje a byla by to pouze ztráta času. Možná někoho může napadnout myšlenka sjednotit Product Ownera do jedné osoby se Scrum Masterem. To je však praktika, která se příliš nedoporučuje. Další možností je sjednotit Scrum

Mastera s Developerem ale ani tato varianta není vhodná, jelikož Scrum Master nebude dostatečně včas připraven odstraňovat případné problémy ostatním developerům, z důvodu vytíženosti zpracovávání položek Product Backlogu. Celkově se tedy vývojový tým skládá z 5 osob + 1 osoba ze strany zadavatele. Nyní, když je jasné, co budeme vyvíjet, a máme k tomu dostupný vývojový tým, se můžeme pustit do první fáze vývoje, kterou je předehra.



Obrázek 8: Tým

5.2.2 Předehra

Zde začíná práce pro Scrum Mastera, jelikož musí zvolit vývojové nástroje, které se při implementaci budou používat. V této úvodní fázi se začíná vytvářet nejdůležitější dokument celého procesu Product Backlog. Tento dokument udržuje po celou dobu vývoje, jelikož do tohoto dokumentu jsou stále přidávány nové požadavky a upravují se stávající. Product Backlog je udržován ve formě listu v programu Microsoft Excel, jehož část si můžete prohlédnout v níže uvedené tabulce. Jelikož je celkový Product Backlog celkem robustní a obsahuje kolem 100 položek je celý tento dokument uveden v přílohách.

Dokument obsahuje co možná nejvíce požadavků vycházejících ze specifikace systému. Každá položka je seřazena podle priority do jednotlivých sprintů a na této prioritě se dohodne zákazník s Product Ownerem. Dále je každé položce přiřazena pracnost daného úkolu. K vybrání pracnosti úkolu bylo použito metody Planning Poker. Při této hře každý dostane karty, kde každá karta symbolizuje počet hodin (král je nekonečno, tedy neproveditelný úkol). Prochází se Product Backlog a každý z týmu udělá na základě detailnějšího popisu úkolu odhad pomocí odhozené karty, která je prozatím zakrytá ostatním členům týmu. Na pokyn se karta otočí a vybírá se člověk s nejvyšším a nejnižším odhadem a ten pak musí podat důvod proč, si myslí, že daný úkol je takto náročný. Dále následuje znova další kolo s tím, že si každý většinou upraví

názor. Pak se zapisuje průměr ze všech odhadů. Tato hra také brání tomu, že si odpovědný vývojář pracnost úkolu podhodnotí nebo nadhodnotí. Scrum Master však tyto hodnoty stále nemůže brát jako finální a musí je tedy brát s rezervou. Toto plánování pracnosti je hlavně jako hrubý odhad avšak už máme jakési povědomí o tom, kolik práce nás čeká a co budeme v blízké budoucnosti dělat.

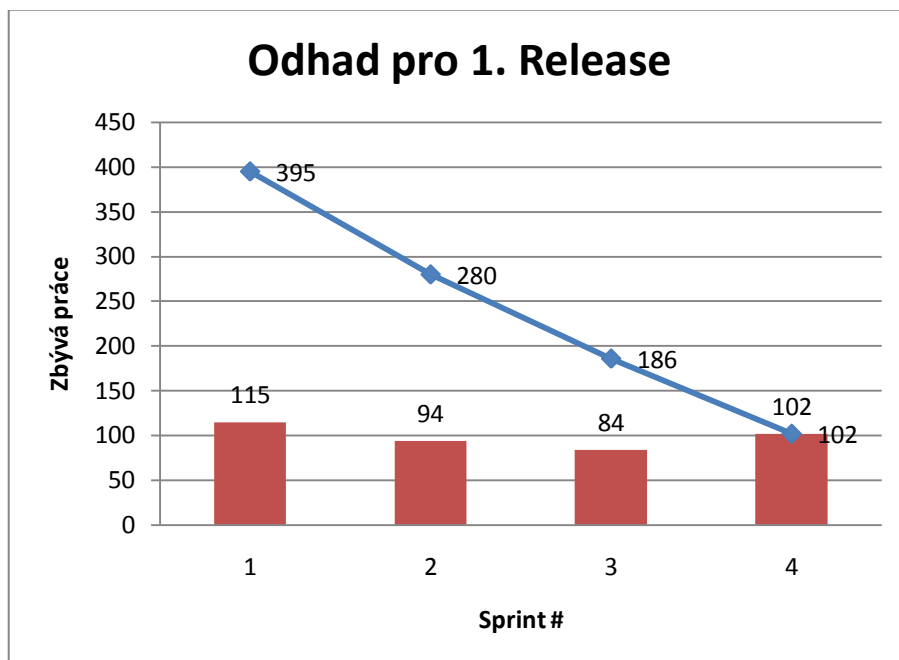
Tabulka 3 uvádí část položek 3., 4. a část 5. sprintu. Takto vypadal dokument v průběhu 4. sprintu. V tabulce je uvedeno jejich jednoznačné identifikační číslo, s inkrementem 1 pro každou další položku. Pokud se nějaká položka ze seznamu odebere, dostane následující položka číslo znovu o jednu větší (nikdy se nepoužívá jedno číslo vícekrát). Dále je zde uveden popis dané položky, který může být formou krátkého slovního popisu nebo formou user story, ve kterém je přesně popsán daný problém. Pro názornost však stačí slovní popis. Poté se nastavuje průběžně status dané položky, která může být typu „*Done*“ pokud je již hotova, „*Removed*“ pokud byla v průběhu vývoje odebrána, „*Ongoing*“ pokud se na položce právě pracuje a nakonec ve stavu „*Planned*“ pokud je tato položka plánována do dalšího sprintu. Následuje odhad pracnosti daného úkolu v každém ze sprintů pomocí určitého čísla. Odhad však nemusí být vždy úplně zřejmý a proto bylo použito speciálních symbolů pro vylepšení odhadování. Symbol „-“ znamená, že daný úkol byl do Product Backlogu přidán až po již proběhlých sprintech. Pokud se nepovedlo odhadnout pracnost úkolu například z důvodu velice rozdílných odhadů, byl takovému úkolu přidělen symbol „?“ a odhad tohoto úkolu se provedl až po uplynutí určitého času, ve kterém bylo možno problém lépe analyzovat a provést lepší odhad. Nad všemi těmito odhady pracnosti byl proveden součet, aby bylo jasné, kolik práce se odhaduje pro první sprint, release a i pro celý Product Backlog. Tato celková data pak využívá i Burn-Down Chart. Jako poslední jsou tu položky, které jsou přeškrtnuté. Jedná se o položky, které byly v průběhu vyškrtnuty a nebudou nadále implementovány. Jako další atribut je v Product Backlogu uvedeno číslo sprintu, do kterého je daná položka naplánována. Posledním důležitým atributem je priorita položky, jejíž pomocí lze snadněji vypořádat, jaké položky by se měli začít implementovat co nejdříve. Ke každé položce je možno přidat také nepovinný komentář.

Product Backlog

Story ID	Story name	Status	Size	Sprint	Priority	Comments
46	Vytvořit nové téma konference	Done	2	3	High	
47	Upravit dané téma konference	Done	2	3	Medium	
48	Smazat téma	Done	1	3	Low	
49	Zobrazit seznam všech termínů konference	Done	2	3	High	
68	Lokalizovat aplikaci pro češtinu	Removed	20	3	Medium	
50	Upravení daného termínu	Done	2	3	Low	
51	Vytvoření termínu	Done	2	3	High	
52	Smazání termínu	Done	1	3	Low	
53	Zobrazit seznam všech HTML stránek	Done	2	3	High	
54	Zobrazit jednotlivé stránky	Ongoing	3	4	High	
55	Vytvořit novou HTML stránku	Ongoing	4	4	High	
56	Upravit danou HTML stránku	Ongoing	4	4	Low	
57	Smazat určitou HTML stránku	Ongoing	1	4	Medium	
61	Správa ubytování a aktivit	Removed	20	4	Medium	
63	Nakonfigurovat SMTP server	Removed	5	4	Low	
58	Zobrazit seznam všech účtů k zaplacení	Ongoing	3	4	High	
59	Zobrazení účtu uživateli	Ongoing	2	4	Medium	
60	Odsouhlasit zaplacení účtu	Ongoing	1	4	Low	
62	Nakonfigurovat server pro IS	Ongoing	15	4	High	
64	Zabezpečit server	Ongoing	4	4	Medium	
65	Nasadit první verzi IS na server	Ongoing	8	4	High	
66	Překlad IP adresy serveru na rtt.vsb.cz	Ongoing	?	4	High	
69	Migrace na jiný server	Removed		4	Medium	
70	Nahrání zálohy systému z důvodu závady na serveru	Ongoing	-	4	High	
71	Vytvořit fotogalerii z konference	Ongoing	-	4	Low	
67	Vytvořit uživatelskou příručku	Planned	30	4	Medium	
72	Zobrazit seznam všech stavů článků	Planned	2	5	High	
73	Vytvořit nový stav článku	Planned	2	5	Medium	
74	Upravit stav článku	Planned	2	5	Low	
75	Smazat stav článku	Planned	1	5	Low	

Tabulka 3: Product Backlog

Dalším artefaktem používaným v metodice Scrum je kromě Product Backlogu i Burn-Down Chart. Obrázek 9 je graf pro první release, který je možno vidět na obrázku uvedeném níže. Bylo naplánováno, že za první sprint se vykoná 115 jednotek práce, za druhý sprint 94 jednotek, za třetí 84 jednotek a ve čtvrtém sprintu 102 jednotek z celkového počtu 395 jednotek práce za celý první release. Po závěrečném sprintu bude uveden Burn-Down Chart pro celý projekt s finálními hodnotami, kde bude možno vyčíst, jak moc se původní odhad liší od finální verze.



Obrázek 9: Graf odhadu pro 1. Release

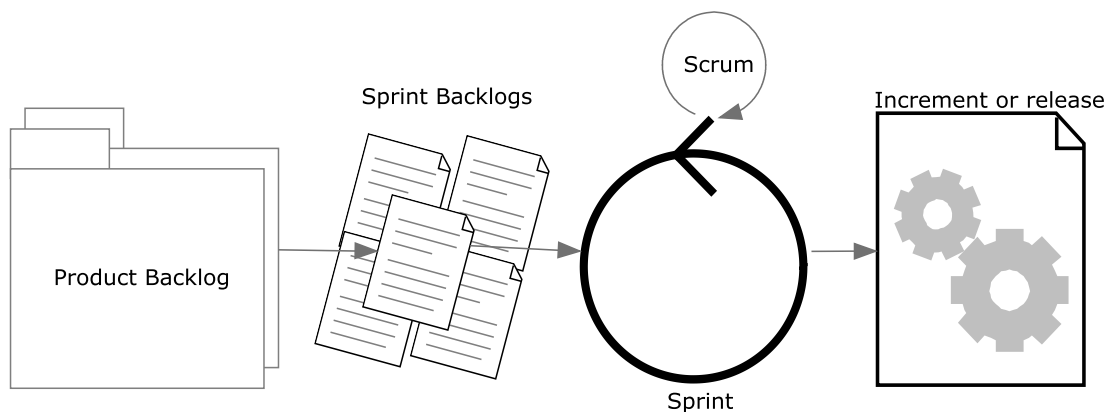
Posledním dokumentem, který byl vytvořen před samotnou implementací, byl Release plan. V tomto plánu se vyskytují plánovaná data vydání jednotlivých releasů a konců sprintů. V případě vyvíjeného konferenčního systému nešlo nijak komplikovaný release plán. Tento plán také slouží jako celková statistická tabulka pro Scrum Mastera počtu plánovaných hodin a důležitých dat, pro plánování případných schůzek. Tento dokument lze ve správně připraveném Product Backlogu také automaticky generovat. Tabulka 4 zobrazuje konkrétní release plán pro vyvíjený projekt ve fázi kdy jsou první tři sprinty hotové, pracuje se na čtvrtém a pátý poslední je naplánován.

Increment Plan							
Incr.	Start	Days	End	Size	Status	Goal	
1	1.5.2010	124	1.9.2010	350			
2	2.9.2010	31	2.10.2010	53			
Sprint Plan							
Sprint	Start	Days	End	Size	Status	Goal	Increment
1	1.5.2010	31	31.5.2010	115	Released	Správa uživatelů je plně funkční	1
2	1.6.2010	31	1.7.2010	94	Released	Uživatelé mohou zasílat své články a lze je recenzovat	1
3	2.7.2010	31	1.8.2010	66	Released	Systém disponuje nejdůležitějšími vlastnostmi	1
4	2.8.2010	31	1.9.2010	75	Ongoing	Hotový systém pro správu konference	1
5	2.9.2010	31	2.10.2010	53	Planned	Systém je plně modifikovatelný a dynamický	2
		</					

Tabulka 4: Release plan

5.2.3 Hra

Tato fáze se skládala z pěti po sobě jdoucích sprintů, ve kterých se implementovaly jednotlivé položky daného Sprint Backlogu. Po prvních čtyřech sprintech bude vydán první release systému, který bude nasazen do prostředí zákazníka. V posledním pátém sprintu se dotvářely zbylé funkce s nejnižší prioritou. Nyní budou popsány jednotlivé sprinty a to jak v nich probíhaly práce. Obrázek 10 ilustruje proces vývoje pomocí metodiky Scrum.



Obrázek 10: Scrum

5.2.3.1 První sprint

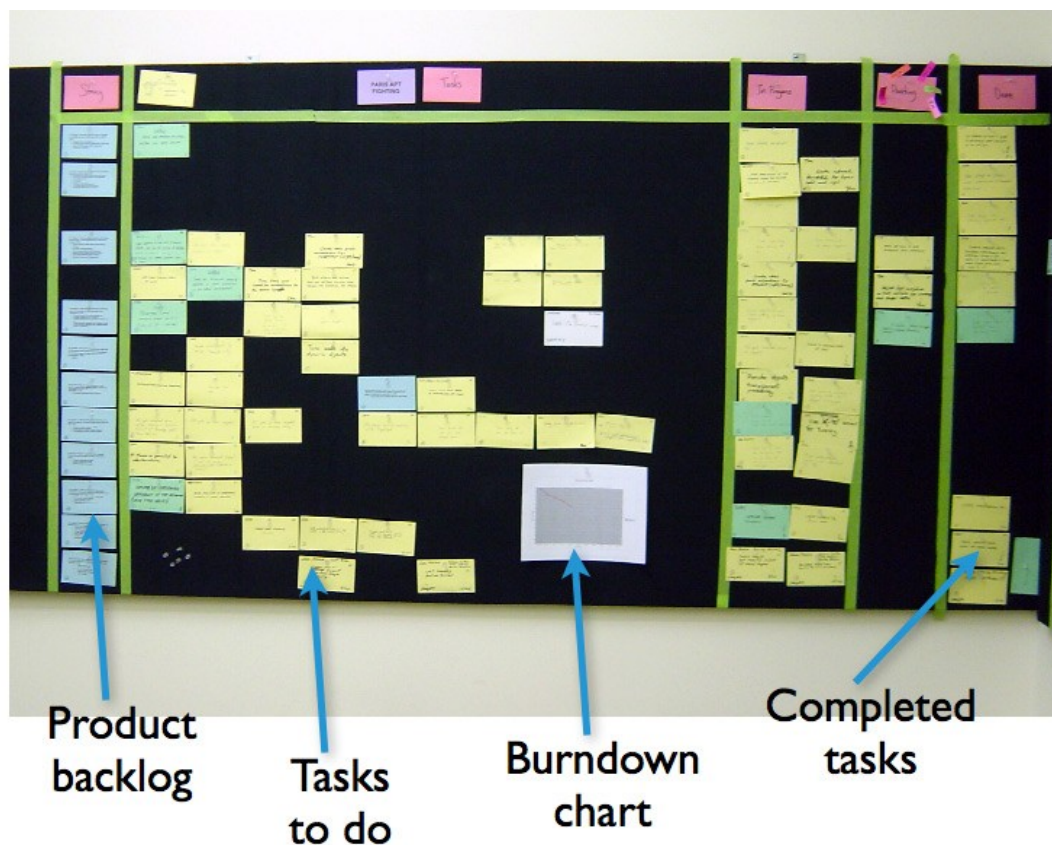
První sprint začal Sprint Planning Meetingem, kde Scrum Master vybral položky s nejvyšší prioritou, které bylo nutné udělat nejdříve. K vybrání těchto položek posloužil připravený

Product Backlog. Když byla vybrána skupina položek, které se budou implementovat v prvním sprintu, byl vytvořen další velice důležitý dokument, kterým byl Sprint Backlog #1. Tento backlog je velice podobný Product Backlogu a tak v něm opět najdete jednotlivé názvy položek, které se budou implementovat a jejich pořadová čísla. Na rozdíl od Product Backlogu se zde ale nově vyskytuje podrobné rozepsání jednotlivých dnů, které sprint zahrnoval. V našem případě měl každý sprint 31 dní. Na začátku sprintu jsou vypsána čísla obsahující startovní hodnotu odhadu pracnosti daného úkolu. Také se zde vyskytuje informace o tom, kdo jakou položku zpracovával. Rozdělení položek probíhalo v demokratickém duchu a díky odhadům pracnosti bylo tomu, kdo měl práce nejméně, direktivně přiděleny položky, o které neměl nikdo zájem. V průběhu sprintu pak Scrum Master zapisoval pokroky ve smyslu odvedené práce na jednotlivých položkách Sprint Backlogu. Kolik práce se na konkrétní položce provedlo, se zjišťovalo pomocí Daily Scrum Meetings, kde se scházel celý vývojový tým. Tato setkání měl na starost Scrum Master a každý den ráno pokládal jednotlivým členům Scrum Teamu otázky „Co jsi dělal včera?“, „Co budeš dělat dnes?“, „Vyskytly se nějaké problémy?“. Díky těmto informacím pak měl možnost pomoci s daným problémem a hlavně přehled o tom, kolik práce ještě zbývá k dokončení. K lepšímu znázornění jednotlivých úkolů sloužila tzv. Task Board, což je tabule, která obsahuje jednotlivé papírky se slovním popisem aktuálního stavu daného úkolu. Pokud někdo přišel na nový úkol, jednoduše vytvořil nový papírek s popisem a přilepil ho na Task Board. Tabulka 5 je příklad konkrétního papírku, který byl použit na vyvíjeném projektu.

Story ID:	2	Story:	Vytvořit datový model		
Task:	Vytvořit datový model				
Responsible Person: Dev					
Initial Estimate	20	Work Done	20	Work Left	0

Tabulka 5: Task

Během nebo před Daily Scrumem se upravovaly odhady a papírky se hýbalo po tabuli. Obrázek 11 představuje příklad, na kterém si lze ukázat, jak takový Task Board vypadá. Více informací lze nalézt na [13].



Obrázek 11: Task Board

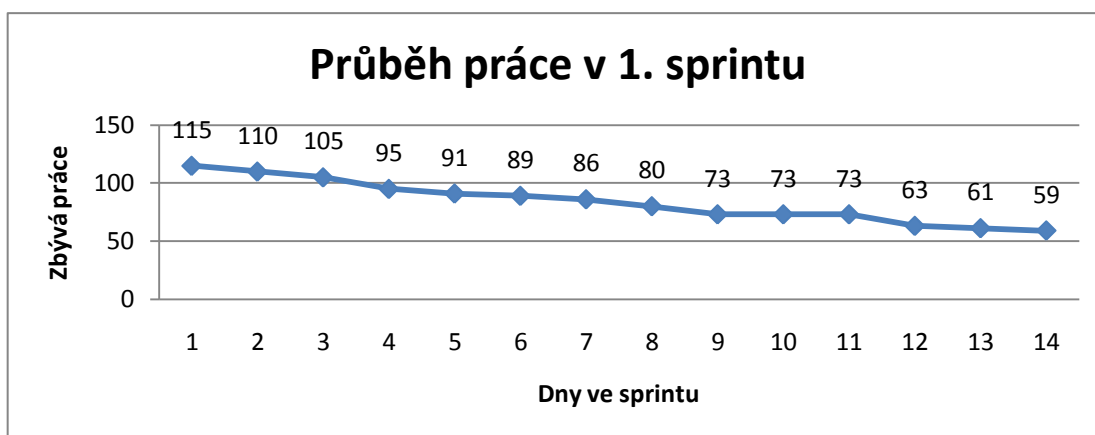
Takto se prováděly schůzky každý den a pracovalo se na projektu až do posledního dne, kdy přichází na řadu Sprint Review Meeting. Na tomto setkání se prezentoval výsledek prvního sprintu zadavateli, který podal zpětnou vazbu. V průběhu tohoto sprintu se nevyskytly problémy a tak se vše stihlo podle plánu. Jelikož do tohoto prvního sprintu nebyly zavedeny náročnější funkce, nebyl tedy problém a tým se nedostal do časového skluzu. Toto setkání bylo spojeno dohromady se Sprint Retrospective, a zde se všichni zúčastnění podělili o dobré i špatné zkušenosti a navrhli zlepšení.

Tabulka 6 představuje část Sprint Backlogu #1. Jak je zřejmé dokument je velice podobný Product Backlogu a jen pro zopakování budou vysvětleny jednotlivé uchovávané atributy. První sloupec obsahuje unikátní ID daného problému v rámci celého Product Backlogu. Ve druhém sloupci je označení člena vývojového týmu, který nese zodpovědnost za daný úkol. Dále je zde slovní popis problému, po němž následuje již nejdůležitější část Scrum Backlogu, kterým je evidence odpracovaných jednotek práce na daném problému. Například u problému číslo 2 je vidět, že v prvním dnu sprintu se na daném úkolu odpracovalo 5 jednotek práce z celkových 20 a v druhém dnu sprintu dalších 5 jednotek. Takto se eviduje každá položka zvlášť. Jelikož by se do textu celý Sprint Backlog nevešel, jsou všechny Sprint Backlogy uvedeny v příloze.

ID	Člen	Úkol	den sprintu / odhad práce													
			1	2	3	4	5	6	7	8	9	10	11	12	13	14
			115	110	105	95	91	89	86	80	73	73	73	63	61	59
1	Dev	Zprovoznit systém v PHP od kolegů z Prahy	10	10	10	10	10	10	10	10	10	10	10	8	6	4
2	Vyv	Vytvořit datový model	20	15	10	10	10	10	10	10	10	10	10	10	10	10
3	Cod	Nakonfigurovat testovací server	6	6	6	6	6	6	6	6	6	6	6	6	6	6
4	Vyv	Přihlášení do systému	4	4	4	0	0	0	0	0	0	0	0	0	0	0
5	Dev	Odhlášení ze systému	4	4	4	4	0	0	0	0	0	0	0	0	0	0
6	Cod	Registrace do systému	8	8	8	8	8	8	8	8	8	8	8	4	4	4
7	Dev	Zaslání zapomenutého hesla	6	6	6	6	6	6	6	6	3	3	3	3	3	3
8	Vyv	Změna hesla	6	6	6	6	6	6	6	3	3	3	3	3	3	3
9	Dev	Zobrazit seznam všech uživatelů systému	2	2	2	2	2	0	0	0	0	0	0	0	0	0
10	Cod	Zobrazit seznam všech uživatelů dopravujících se autobusem	3	3	3	3	3	3	0	0	0	0	0	0	0	0
11	Dev	Zobrazit seznam všech uživatelů, kteří se dostaví na konferenci	3	3	3	3	3	3	3	0	0	0	0	0	0	0
12	Vyv	Zobrazit profil daného uživatele	4	4	4	4	4	4	4	4	4	4	4	0	0	0
13	Vyv	Upravit profil daného uživatele	4	4	4	4	4	4	4	4	4	4	4	4	4	4
14	Cod	Smazání uživatele	5	5	5	5	5	5	5	5	5	5	5	5	5	5
15	Cod	Vybrat free hosting pro ASP aplikace	2	2	2	2	2	2	2	2	2	2	2	2	2	2
16	Vyv	Nakonfigurovat ukázkovou verzi aplikace pro spuštění na free hostingu	8	8	8	8	8	8	8	8	8	8	8	8	8	8
17	Dev	Vytvořit základní design systému	20	20	20	14	14	14	14	14	10	10	10	10	10	10

Tabulka 6: Sprint Backlog #1

Obrázek 12 uvádí Burn-Down Chart patřící prvnímu Sprint Backlogu, který byl vytvořen pro názornější představu o hotové práci. Protože se však v prvním Sprintu neobjevily žádné problémy, je tento graf velmi jednoduchý a zřídka kdy se stává tento ideální stav. I tyto grafy jsou všechny uvedeny v přílohách, jelikož by se celé do textu nevešly.



Obrázek 12: Burn-Down Chart Sprint #1

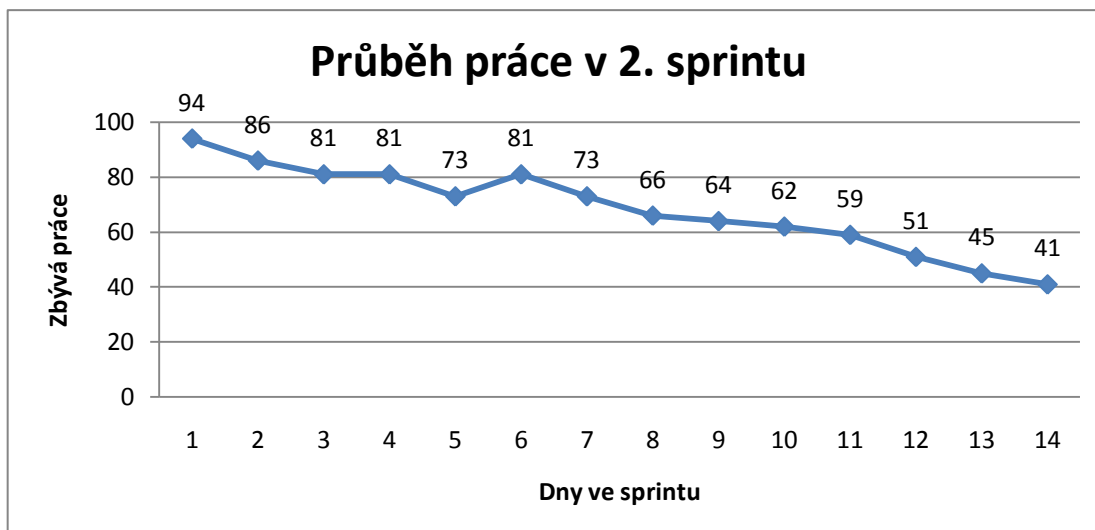
5.2.3.2 Druhý sprint

Do druhého sprintu tedy tým vstupoval s čistým štítem a mohla tak v klidu proběhnout druhá Sprint Planning schůzka. Opět byla vybrána množina položek a jednotliví vývojáři si vybrali podmnožinu položek, které měli vypracovat. Po této schůzce byl vytvořen Sprint Backlog #2, do kterého byl v průběhu sprintu zapisován vývoj odvedené práce, což bylo průběžně zjišťováno na každodenních ranních schůzkách týmu. V tomto sprintu se vyskytl problém, při bližší analýze položky zabývající se vytvořením algoritmu vyhodnocení stavu článku. Ukázalo se, že tato položka byla při prvním ohodnocování podhodnocena a tak byla v 6. dni 2. sprintu změněna hodnota pracnosti tohoto úkolu z 15 na 25. Tabulka 7 znázorňuje dané změny ve Sprint Backlogu. O takovýto nárůst pracnosti se zasloužily zejména změny tohoto algoritmu zadané přímo zákazníkem, který neměl stále úplně přesnou představu o tom, jak se budou automaticky články vyhodnocovat.

Obrázek 13 ukazuje, že se změna samozřejmě promítla i do Burn-Down Chartu. Kromě této události se ve sprintu nevyskytly další problémy. Na následném Sprint Review Meetingu tak vše proběhlo v pořádku a zákazník byl obeznámen a spokojen se současným stavem a vývojem aplikace.

ID	Člen	Úkol	den sprintu / odhad práce													
			1	2	3	4	5	6	7	8	9	10	11	12	13	14
			94	86	81	81	73	81	73	66	64	62	59	51	45	41
18	Dev	Seřadit články podle tématu	6	6	6	6	6	6	6	6	6	6	6	6	6	6
19	Cod	Zobrazit články daného uživatele	4	4	4	4	4	4	4	4	4	4	4	4	4	0
20	Vyv	Zobrazit seznam všech článků v konferenci	2	2	2	2	2	2	2	2	2	2	2	2	2	2
21	Dev	Zobrazit detaily článku	4	4	4	4	4	4	4	4	4	4	4	4	0	0
22	Cod	Upload článku	8	0	0	0	0	0	0	0	0	0	0	0	0	0
23	Cod	Vytvořit nový článek	10	10	5	5	0	0	0	0	0	0	0	0	0	0
24	Dev	Upravit článek	10	10	10	10	7	5	0	0	0	0	0	0	0	0
25	Cod	Smazat článek	4	4	4	4	4	4	4	0	0	0	0	0	0	0
26	Vyv	Zaslat finální verzi článku	10	10	10	10	10	10	8	6	5	5	5	0	0	0
27	Vyv	Zobrazit komentáře z recenzí k článku	3	3	3	3	3	3	3	3	3	3	3	3	3	3
28	Vyv	Přiřadit recenzenta danému článku	1	1	1	1	1	1	1	1	1	1	1	1	1	1
29	Vyv	Vytvořit novou recenzi	1	1	1	1	1	1	1	1	1	1	1	1	1	1
30	Dev	Zobrazit seznam všech recenzí	2	2	2	2	2	2	2	2	2	2	2	2	2	2
31	Cod	Zobrazit seznam recenzí pro daného recenzenta	2	2	2	2	2	2	2	2	2	2	2	2	2	2
32	Dev	Recenzovat článek	8	8	8	8	8	8	8	8	8	8	8	8	8	8
33	Cod	Přeřadit recenzi jiného recenzenta	2	2	2	2	2	2	2	2	2	2	2	2	2	2
34	Dev	Smazat recenzi	2	2	2	2	2	2	2	2	2	2	2	2	2	2
35	Vyv	Algoritmus vyhodnocení stavu článku	15	15	15	15	15	25	24	23	22	20	17	14	12	12

Tabulka 7: Sprint Backlog #2

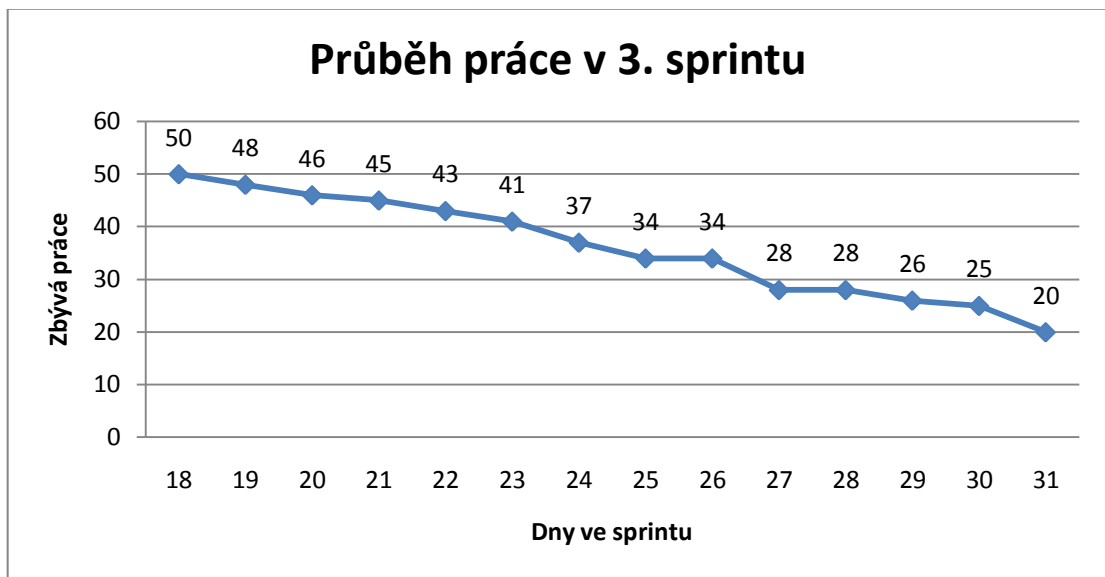


Obrázek 13: Burn-Down Chart Sprint #2

5.2.3.3 Třetí sprint

Nejspíše z důvodu klidného běhu vývoje produktu nebo z důvodu, že zadavatel až v průběhu vývoje přichází na to, co vlastně od výsledku očekává, začal zadavatel vytvářet a podstatně měnit požadavky na systém. Tyto nové požadavky bylo nutno doplnit do Product Backlogu a následně přidat co nejdříve do dalšího sprintu. Na Sprint Planning Meetingu byly jednotlivé funkce rozděleny do týmu a byl vytvořen Sprint Backlog #3. Tabulka 8: Sprint Backlog #3 reprezentuje část tohoto dokumentu. V tomto sprintu bylo myšleno kvůli změnám na problémy, a proto nebylo do tohoto sprintu vybráno tolik položek z Product Backlogu jako bylo v minulých sprintech, ale z důvodu klidnějších prvních dvou sprintů bylo možno si toto odlehčení dovolit. V průběhu sprintu přišla od zadavatele důležitá změna, a jelikož se Scrum řídí pravidlem, že veškeré nové požadavky se přidávají nejdříve do příštího sprintu, chtěl Scrum Master aby byla změna zavedena až do dalšího sprintu. Při další ranní schůzce se však tým domluvil, na porušení tohoto pravidla a změna bude zavedena již do tohoto sprintu. Tímto způsobem však narostl objem práce, kterou bylo potřeba naimplementovat v daném sprintu. Pečlivě připravený Sprint Backlog je však na tyto náhlé změny připraven a zde se také ukazují výhody agilních metodik a jejich flexibilita. Tabulka 8 zobrazuje Sprint Backlog v druhé polovině vývoje, tedy od 18. – 31. dne sprintu. Danou změnou ve vývoji byl požadavek na lokalizaci systému do českého jazyka. Jelikož je podpora dalších jazyků velice náročný požadavek hlavně na čas překládání jednotlivých textů, měla se tomuto požadavku nastavit vysoká priorita a začít na lokalizaci pracovat co nejdříve. Tento náročný požadavek však rozhodil plán vývoje tak značně, že tento požadavek nakonec ve třetím sprintu nebyl splněn. V tomto případě musel tuto situaci vysvětlit Scrum Master zákazníkovi na Sprint Review Meetingu, který nebyl z této situace příliš nadšený ale, jelikož se jednalo o náhlou změnu, byl nakonec zákazník po prezentaci ostatních vyvíjených funkcí spokojený a souhlasil s přenesením požadavku lokalizace do dalšího sprintu. Obrázek 14 umožňuje shlédnout vývoj práce, kde je

vidět, že na konci sprintu nebyla hodnota zbylé práce 0, jako je v případě úspěšně dokončeného sprintu, ale 20, což odpovídá hodnotě požadavku vypracování lokalizace.



Obrázek 14: Burn-Down Chart Sprint #3

ID	Člen	Úkol	den sprintu / odhad práce													
			18	19	20	21	22	23	24	25	26	27	28	29	30	31
			50	48	46	45	43	41	37	34	34	28	28	26	25	20
36	Vyv	Zaslání emailu o stavu článku	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37	Vyv	Zobrazit seznam všech otázek	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38	Vyv	Upravit vybranou otázku	0	0	0	0	0	0	0	0	0	0	0	0	0	0
39	Vyv	Vybrat vhodnou šablonu pro vzhled a upravit jí	10	10	10	10	8	8	6	4	4	4	4	4	4	0
68	Dev	Lokalizovat aplikaci pro češtinu	20	20	20	20	20	20	20	20	20	20	20	20	20	20
40	Cod	Lokalizovat aplikaci pro angličtinu	8	8	8	8	8	8	8	8	8	2	2	2	1	0
41	Cod	Zobrazit všechny druhy dopravy na konferenci	0	0	0	0	0	0	0	0	0	0	0	0	0	0
42	Cod	Vytvořit novou možnost dopravy na konferenci	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	Cod	Upravit možnost dopravy	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44	Cod	Smazat možnost dopravy	0	0	0	0	0	0	0	0	0	0	0	0	0	0
45	Cod	Zobrazit seznam všech témat konference	2	0	0	0	0	0	0	0	0	0	0	0	0	0
46	Cod	Vytvořit nové téma konference	0	0	0	0	0	0	0	0	0	0	0	0	0	0
47	Vyv	Upravit dané téma konference	2	2	0	0	0	0	0	0	0	0	0	0	0	0
48	Vyv	Smazat téma	1	1	1	0	0	0	0	0	0	0	0	0	0	0
49	Dev	Zobrazit seznam všech termínů konference	2	2	2	2	2	0	0	0	0	0	0	0	0	0
50	Dev	Upravení daného termínu	2	2	2	2	2	2	0	0	0	0	0	0	0	0
51	Dev	Vytvoření termínu	2	2	2	2	2	2	2	2	2	2	2	0	0	0
52	Dev	Smazání termínu	1	1	1	1	1	1	1	0	0	0	0	0	0	0

Tabulka 8: Sprint Backlog #3

5.2.3.4 Čtvrtý sprint

Víc než polovina položek byla již hotova a hlavní části systému byly již funkční, a tak se ve čtvrtém sprintu na Sprint Planning Meetingu vybíraly zbylé položky z Product Backlogu z čehož vznikl Sprint Backlog #4. V průběhu tohoto sprintu bylo pár požadavků vyškrtáno po konzultaci se zákazníkem, takže práce ubývala vývojářům nejen díky jejich pracovitosti ale také změnám ve Sprint Backlogu. Zajímavostí, která nastala v tomto sprintu, byl výpadek serveru z důvodu letní bouřky. Při této bouři se neznámým způsobem nahrála na testovací server stará verze systému včetně zdrojových kódů. I na tuto situaci však dokáže Scrum reagovat a tak bylo

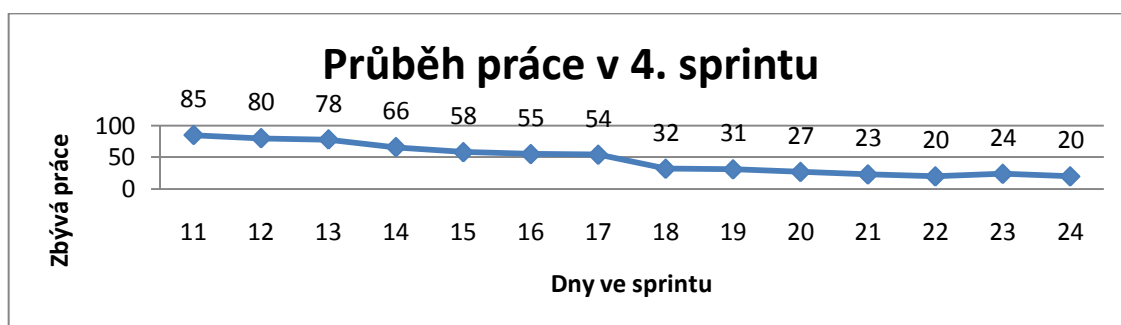
potřeba tento problém vyřešit. Tabulka 9 znázorňuje položky, které byly v průběhu sprintu odstraněny. Mezi těmito položkami byl i úkol lokalizace aplikace do češtiny, který se nestihl v minulém sprintu. Vyškrtnuté položky jsou vyznačeny přeškrtnutým textem v tabulce. Rozdíl oproti sprintům minulým lze pozorovat u položky 71, u které je patrné, že požadavek přišel až ve 24. dni sprintu.

ID	Člen	Úkol	den sprintu / odhad práce													
			11	12	13	14	15	16	17	18	19	20	21	22	23	24
			85	80	78	66	58	55	54	32	31	27	23	20	24	20
53	Vyv	Zobrazit seznam všech HTML stránek	2	2	2	2	2	2	2	2	2	2	2	2	0	0
54	Vyv	Zobrazit jednotlivé stránky	3	3	3	3	3	3	3	3	3	3	3	0	0	0
55	Dev	Vytvořit novou HTML stránku	4	4	4	4	4	4	4	4	4	4	0	0	0	0
56	Dev	Upravit danou HTML stránku	4	4	4	4	4	4	4	4	4	0	0	0	0	0
57	Dev	Smazat určitou HTML stránku	1	1	1	1	1	1	1	1	0	0	0	0	0	0
58	Cod	Zobrazit seznam všech účtů k zaplacení	0	0	0	0	0	0	0	0	0	0	0	0	0	0
59	Dev	Zobrazení účtu uživateli	0	0	0	0	0	0	0	0	0	0	0	0	0	0
68	Dev	Lokalizovat aplikaci pro češtinu	20	20	20	0	0	0	0	0	0	0	0	0	0	0
60	Dev	Odsouhlasit zaplacení účtu	0	0	0	0	0	0	0	0	0	0	0	0	0	0
61	Vyv	Správa ubytování a aktivit	20	20	20	20	20	20	20	-	-	-	-	-	-	-
62	Vyv	Nakonfigurovat server pro IS	0	0	0	0	0	0	0	0	0	0	0	0	0	0
63	Dev	Nakonfigurovat SMTP server	5	-	-	-	-	-	-	-	-	-	-	-	-	-
64	Cod	Zabezpečit server	0	0	0	0	0	0	0	0	0	0	0	0	0	0
65	Dev	Nasadit první verzi IS na server	8	8	8	8	8	8	8	8	8	8	8	8	8	8
66	Vyv	Překlad IP adresy serveru na rtt.vsb.cz	-	-	-	-	-	-	-	-	-	-	-	-	-	-
69	Cod	Migrace na jiný server	-	-	-	?	?	?	?	?	?	?	-	-	-	-
70	Dev	Nahrání zálohy systému z důvodu závady na serveru	-	-	-	8	0	0	0	0	0	0	0	0	0	0
71	Vyv	Vytvořit fotogalerii z konference	-	-	-	-	-	-	-	-	-	-	-	-	8	4
67	Cod	Vytvořit uživatelskou příručku	18	18	16	16	16	13	12	10	10	10	10	10	8	8

Tabulka 9: Sprint Backlog #4

Druhým zajímavým jevem může být položka 69, na kterou přišel požadavek až v průběhu sprintu a navíc nebyl k jejímu řešení vybrán konkrétní hrubý odhad. Nelze se tedy divit, že tato položka byla nakonec ze sprintu vyškrtnuta. Díky úbytku práce z důvodu škrtnů ve Sprint Backlogu, byla časová rezerva pro vyřešení všech položek. Na následujícím Sprint Review Meetingu se již prezentovala první verze systému, která byla předána zákazníkovi jako 1. Release. Systém byl nahrán na server zákazníka a nastal první oficiální běh systému.

Obrázek 15 pozoruje poměrně klidný průběh práce ve sprintu. Pouze v případě 17. dne sprintu je možno pozorovat dramatický úbytek práce z důvodu škrtnů ve Sprint Backlogu. Tak velký počet práce by totiž jinak nebylo možné stihnout za jediný den. Lze ještě pozorovat přírůstek práce v 23. dni sprintu, kdy byl přidán požadavek na vytvoření fotogalerie.



Obrázek 15: Burn-Down Chart Sprint #4

5.2.3.5 Závěrečný sprint

Poslední pátý sprint se již zabýval úplně posledními požadavky a změnami, na které přišel zákazník po zkušenostech, které již nabývali cíloví uživatelé. Jelikož v posledním sprintu již nebylo příliš náročné práce, vyvíjeli se položky s nejnižší prioritou a nevyskytly se ani žádné problémy, které bych zde již nepopsal, byla na posledním Sprint Review Meetingu prezentována finální verze zákazníkovi. Tímto dnem byly zpracovány všechny položky jak Sprint Backlogu tak i Product Backlogu a byl vytvořen druhý a poslední release systému. Související Sprint Backlog #5 a jeho Burn-Down Chart lze nalézt v přílohách práce.

5.2.4 Dohra

Tato poslední fáze probíhá již lineárně. Výsledný systém byl integrován přes vzdálenou plochu na server zákazníka. Metodika říká, že dokumentace se vytváří až v této fázi. V našem ukázkovém projektu se však dokumentace začala, vytvářela již ve čtvrtém sprintu. Po této zkušenosti však doporučuji vážně vytvářet dokumentaci až v této poslední fázi. Jako úplně posledním krokem, který v této práci k metodice Scrum je testování. Jelikož metodika přímo nepředepisuje, jak se má výsledný produkt testovat, přistoupili jsme k jednotkovým testům, funkcionálnímu a k manuálnímu testování.

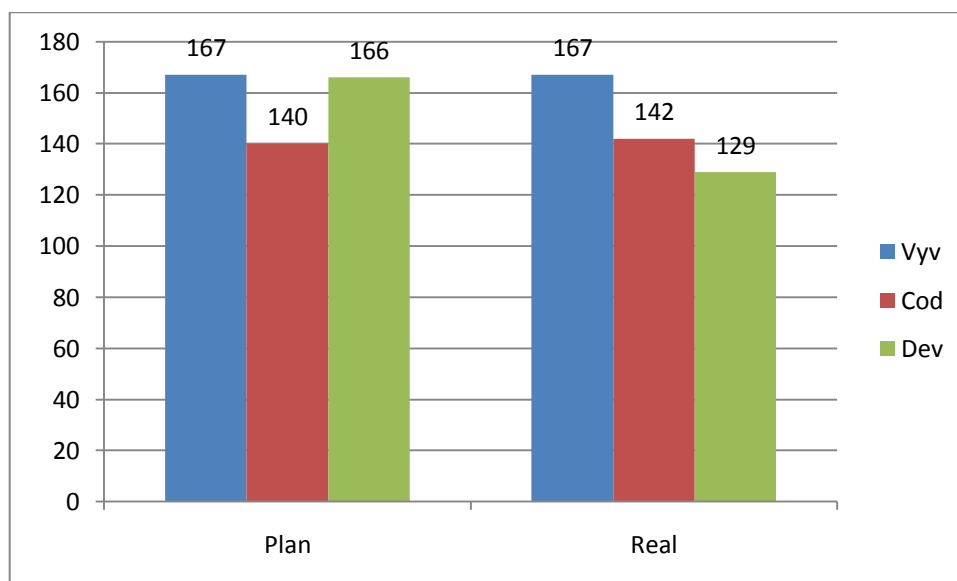
5.3 Velocita týmu

Tabulka 10 znázorňuje celkové statistiky odhadnutých a oproti nim, odpracovaných jednotek práce jednotlivým vývojářům v týmu.

		Sprint #					Sum
		1.	2.	3.	4.	5.	
Vyv	Plan	46	32	29	40	20	167
	Real	46	42	29	30	20	167
Cod	Plan	24	30	28	37	21	140
	Real	24	30	28	37	23	142
Dev	Plan	45	32	27	45	17	166
	Real	45	32	7	28	17	129

Tabulka 10: Velocita týmu

Obrázek 16 ukazuje, jaký byl původní odhad práce přidělené každému vývojáři zvlášť a v druhé polovině grafu lze pozorovat skutečné hodnoty odpracovaných jednotek práce. První vývojář označený jako „Vyv“ podle grafu odpracoval stejný obnos práce, jako ten který mu byl odhadnut. Druhý vývojář odpracoval nakonec o dvě jednotky práce víc, než mu bylo odhadnuto a poslední vývojář „Dev“ měl naplánováno více práce, než nakonec doopravdy odpracoval. Díky podrobné dokumentaci v této metodice tak má možnost Scrum Master také zjistit, kdo se na projektu nejvíce podílel a odvedl největší kus práce na finálním produktu. Lze takto také identifikovat programátory, kteří se do projektu příliš nezapojují a řešit tak tuto situaci.

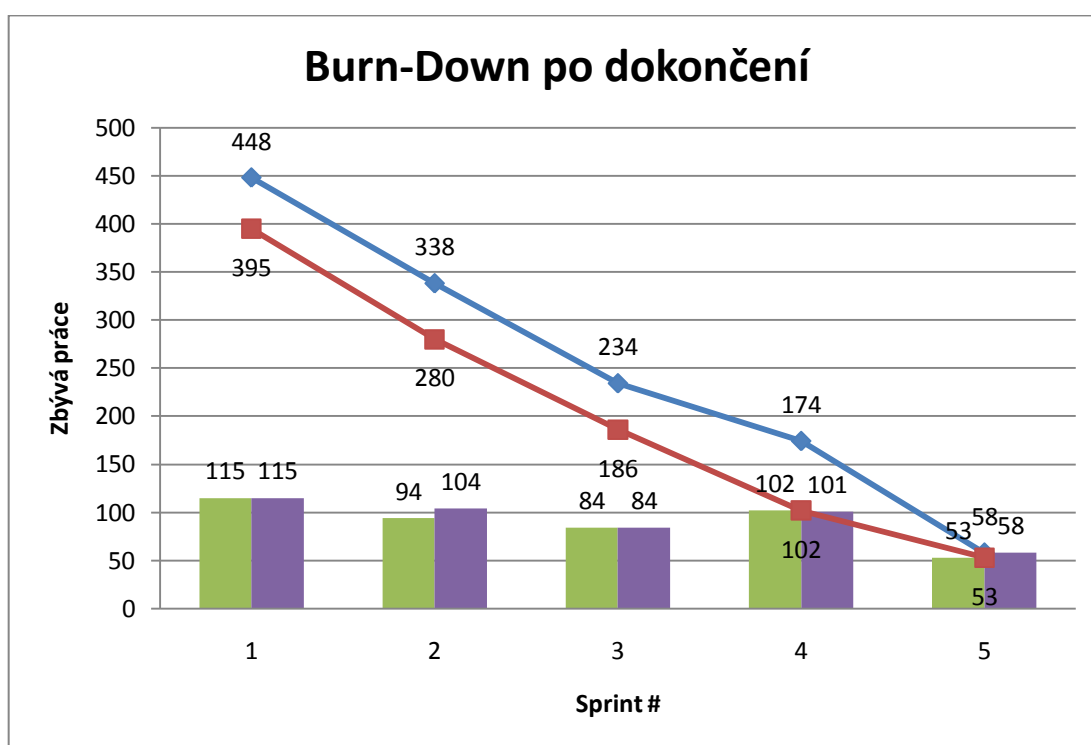


Obrázek 16: Graf výkonnosti vývojářů

5.4 Hodnocení metodiky

Na případové studii je dobře vidět, že metodika se zaměřuje hlavně na manažerské řízení a organizaci projektu. Pro velikost nebo spíše „malost“ ilustračního projektu bylo jistě využití Scrumu vhodné. V průběhu sprintů bylo možno vidět, jak rychle se dokáže Scrum přizpůsobit nepředvídané změně. V prvním sprintu je znázorněný ideální případ vývoje, který ovšem není příliš častý, ale pro zahajovací sprint je jistě vhodné dodat zákazníkovi vše, na čem se s ním výrobce dohodne. Zvyšuje to důvěru zákazníka k dodavateli, že si vybral výrobce správně. Na dalších sprintech bylo ihned vidět, pokud se pracovalo s určitou změnou v plánu.

Obrázek 17 znázorňuje změny oproti počátečnímu odhadu. Červeným spojnicovým grafem je zobrazena křivka původního odhadu a modře reálný graf hodnot. Zeleným sloupcovým grafem je znázorněno, kolik práce bylo odhadnuto, že bude odpracováno a fialovým grafem hodnoty, které se doopravdy v reálném projektu odpracovaly. Celkově práce na celém projektu zabrala 448 jednotek z čehož pro první release to bylo 404 jednotek práce. Průměrně se v každém sprintu odvedlo 89,6 jednotek práce, a každý den bylo odpracováno 2,89 jednotek z konečného seznamu položek. Na grafu lze lehce pozorovat, že díky rychlému zachycení změn se výsledné hodnoty příliš neodlišují od hodnot odhadnutých na začátku.



Obrázek 17: Burn-Down Chart Odhad vs. Realita

6 Závěr

Cílem této práce bylo seznámit se s konferenčními systémy a jejich vlastnosti dále implementovat do konkrétního konferenčního systému vyvíjeného na míru zákazníkovi. Výsledkem tohoto úsilí je konferenční systéme RTT, který byl využit již při konání jedné reálné konference. Systém je připraven pro další vývoj, aby pokrýval stále více funkcí, které s konferenčními systémy souvisí.

Dalším krokem v této práci bylo získat přehled v oblasti agilních metodik vývoje software a popsání jednoho vybraného zástupce. Pro lepší vývoj této aplikace bylo využito agilní metodiky SCRUM, která pomohla s naplánováním celého vývoje. Celý tento plánovací proces byl vysvětlen a podrobně popsán včetně osobních zkušeností a doporučení, které byly při reálném využití metodiky získány.

Díky této práci jsem si na vlastní kůži vyzkoušel tvorbu systému od počátku, kdy jsem každý den zpracovával nové a nové požadavky, což mě často dostávalo do situací, kdy jsem musel přesně pochopit, co po mě vlastně zákazník chce a ne vždy je toto tak jednoduché jak se možná může někomu jevit. Za dobrou zkušenost také pokládám, že jsem si vyzkoušel jaké to je jednat s reálným zákazníkem. Nejzajímavější na práci však bylo použití agilní metodiky pro vývoj aplikace. Při použití metodiky SCRUM jsem zjistil, že vývoj větších systémů podle mého názoru potřebuje pevný řád, podle kterého se pak daný produkt vyvíjí. Za finálním produktem tak nestojí pouze napsání zdrojových kódů ale také potřebný management času, práce, lidí atp. Myslím, že agilní metodiky mají slibnou budoucnost i nadále a budou stále více a více využívány ať už v klasickém znění jak je známe dnes, nebo v kombinacích několika metodik najednou. Samotná metodika SCRUM má dnes spousty příznivců ale také odpůrců. V každém případě se já budu počítat mezi příznivce a těším se na další projekt vedený pomocí této metodiky.

Tato práce tak může v budoucnu sloužit jako příručka pro vývojáře, kteří budou chtít při vývoji náročnějšího a komplexnějšího systému využít agilní metodiky SCRUM.

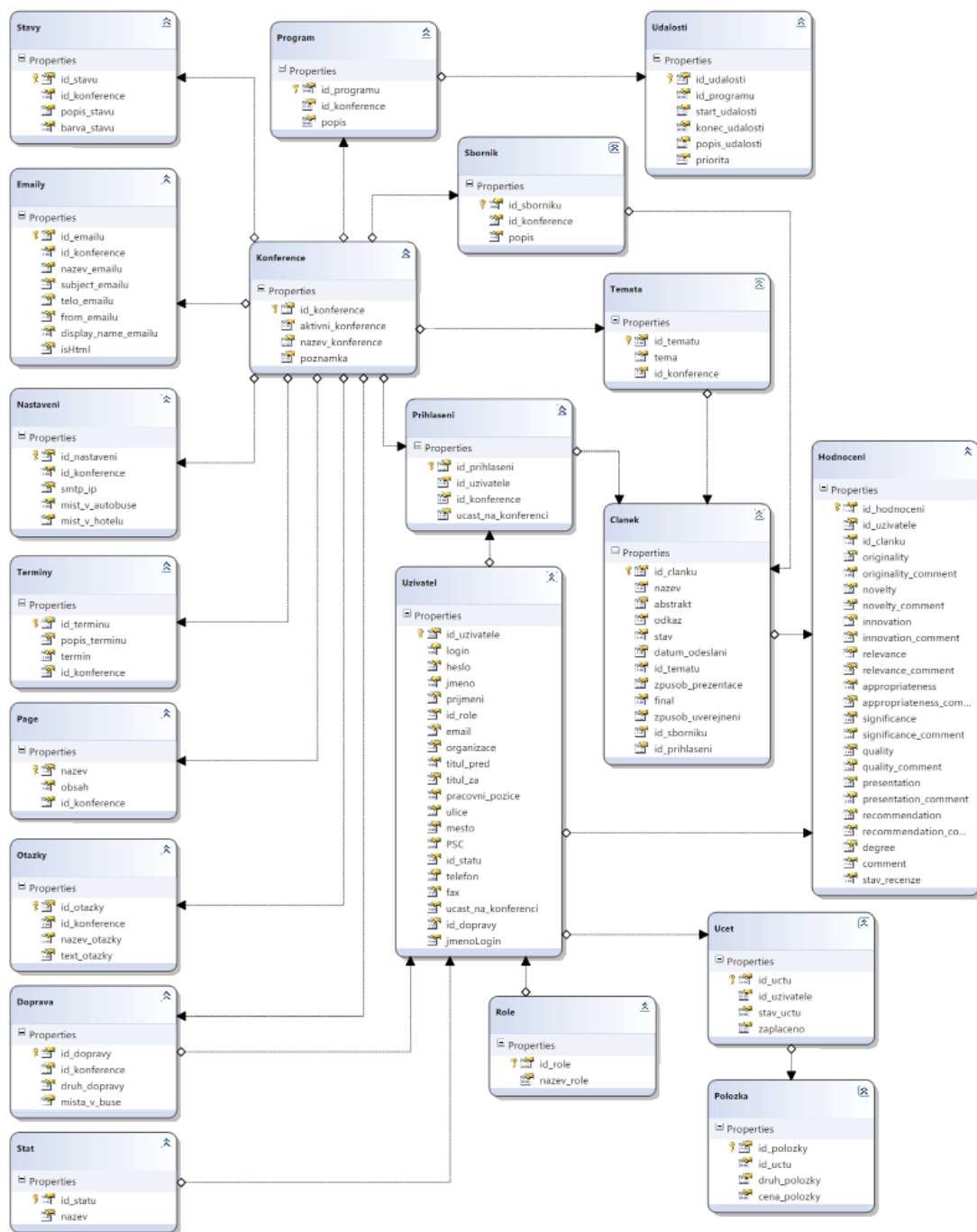
Literatura

- [1] ŠÚST, Ján. *IT podpora organizačních procesů vědeckých konferencí* [online]. Brno, 2009. 69 s. Diplomová práce. Masarykova univerzita, Fakulta informatiky. Dostupné z WWW: <http://is.muni.cz/th/99126/fi_m_b1/DIPLOMOVA-PRACA.pdf>.
- [2] *Public Knowledge Project* [online]. 2011 [cit. 2011-04-16]. Open Conference Systems. Dostupné z WWW: <<http://pkp.sfu.ca/?q=ocs>>.
- [3] *EasyChair Conference System* [online]. 2011 [cit. 2011-04-16]. EasyChair Conference System. Dostupné z WWW: <<http://www.easychair.org/>>.
- [4] *OpenConf* [online]. 2011 [cit. 2011-04-16]. OpenConf Conference Management Peer-Review Software & Hosting Service. Dostupné z WWW: <<http://www.openconf.com/>>.
- [5] *Conference Review System* [online]. 2011 [cit. 2011-04-16]. Conference Review System. Dostupné z WWW: <<http://www.conferencereview.com/index.asp>>.
- [6] KOTTNAUER, Jakub. Jemný úvod do teorie ASP.NET MVC. *Programujte.com* [online]. 2009, [cit. 2011-04-16]. Dostupný z WWW: <<http://programujte.com/?akce=clanek&cl=2009050200-jemny-uvod-do-teorie-asp-net-mvc>>.
- [7] HORÁK, Jan. Scrum - metoda agilního programování. *IT koutek* [online]. 2010, [cit. 2011-04-16]. Dostupný z WWW: <<http://www.wild-web.eu/blog/scrum-metoda-agilni-programovani/>>.
- [8] KNEŠL, Jiří. *Zdroják.cz* [online]. 2009 [cit. 2011-04-16]. Agilní vývoj: Scrum. Dostupné z WWW: <<http://zdrojak.root.cz/clanky/agilni-vyvoj-scrum/>>.
- [9] HAJDIN, Tomáš. *Agilní metodiky vývoje software* [online]. Brno, 2005. 71 s. Diplomová práce. Masarykova univerzita, Fakulta informatiky. Dostupné z WWW: <http://is.muni.cz/th/39440/fi_m/dp.pdf>.
- [10] ŠOCHOVÁ, Zuzana. *Zuzi's blog* [online]. 2011 [cit. 2011-04-16]. Kdo je to Scrum Master? A kdo Product Owner?. Dostupné z WWW: <<http://soch.cz/blog/management/agile/scrum-management/kdo-je-to-scrum-master-a-kdo-product-owner>>.
- [11] ZIKMUND, Martin. *BusinessVize.cz* [online]. 2010 [cit. 2011-04-16]. Agilní projektové řízení - novinka stará přes 20 let. Dostupné z WWW: <<http://www.businessvize.cz/rizeni-a-optimalizace/agilni-projektove-rizeni>>.

- [12] BOSÁK, Martin. *Agilní přístup v projektovém řízení* [online]. Praha, 2008. 58 s. Bakalářská práce. Vysoká škola ekonomická, Fakulta informatiky a statistiky. Dostupné z WWW: <<http://www.martinbosak.cz/files/BP%20-%20Martin%20Bosak%20-%20Agilni%20pristup%20v%20PM.pdf>>.
- [13] ARLOW, Jim; NEUSTADT, Ila. *UML2 a unifikovaný proces vývoje aplikací*. Brno : Computer Press, a.s., 2008. 567 s.
- [14] *Mountain Goat Software* [online]. 2011 [cit. 2011-04-16]. Training for Scrum Task Board Use. Dostupné z WWW: <<http://www.mountaingoatsoftware.com/scrum/task-boards>>.
- [15] *Google Analytics* [online]. 2011 [cit. 2011-04-16]. Google Analytics. Dostupné z WWW: <<http://www.google.com/analytics/>>.



Přílohy



Příloha č. 1: ER Diagram







Příloha č. 2: Lineární zápis typů entit



Legenda:



- Primární klíč: 
- Cizí klíč: 

Stavy ( id_stavu,  id_konference, popis_stavu, barva_stavu)

Emaily ( id_emailu,  id_konference, nazev_emailu, subject_emailu, telo_emailu, from_emailu, display_name_emailu, isHtml)


Nastaveni ( id_nastaveni,  id_konference, smtp_ip, mist_v_autobuse, mist_v_hotelu)



Terminy ( id_terminu, popis_terminu, termin,  id_konference)


Page ( nazev, obsah,  id_konference)



Otazky ( id_otazky,  id_konference, nazev_otazky, text_otazky)





Doprava ( id_dopravy,  id_konference, druh_dopravy, mista_v_buse)




Stat ( id_statu, nazev)



Program ( id_programu,  id_konference, popis)



Konference ( id_konference, aktivni_konference, nazev_konference, poznamka)





Sbornik ( id_sborniku,  id_konference, popis)

Prihlaseni ( id_prihlaseni,  id_uzivatele,  id_konference, ucast_na_konferenci,  id_dopravy)

Uzivatel ( id_uzivatele, login, heslo, jmeno, prijmeni,  id_role, email, organizace, titul_pred, titul_z, pracovni_pozice, ulice, město, PSC,  id_statu, telefon, fax, jmenoLogin)

Udalosti ( id_udalosti,  id_programu, start_udalosti, konec_udalosti, popis_udalosti, priorita)

Temata ( id_tematu, tema,  id_konference)

Clanek ( id_clanku, nazev, abstrakt, odkaz, stav, datum_odeslani,  id_tematu, zpusob_prezentace, final, zpusob_uverejneni,  id_sborniku,  id_prihlaseni)

Role (🔑 id_role, nazev_role)

Hodnoceni (🔑 id_hodnoceni, 🧑 id_uzivatele, 🧑 id_clanku, originality, originality_comment, novelty, novelty_comment, innovation, innovation_comment, relevance, relevance_comment, appropriateness, appropriateness_comment, significance, significance_comment, quality, quality_comment, presentatiton_comment, recommendation, recommendation_comment, degrese, cement, stav_recenze)

Ucet (🔑 id_uctu, 🧑 id_uzivatele, stav_uctu, zaplaceno)

Polozka (🔑 id_polozky, 🧑 id_uctu, druh_polozky, cena_polozky)

Příloha č. 3: Lineární zápis typů vazeb

STAVY_CLANKU_NA_KONFERENCI (Konference, Stavy) 1:N

EMAILY_NA_KONFERENCI (Konference, Emaily) 1:N

NASTAVENI_KONFERENCE (Konference, Nastaveni) 1:N

TERMINY_KONFERENCE (Konference, Terminy) 1:N

STRANKY_KONFERENCE (Konference, Page) 1:N

OTAZKY_V_RECENZICH_KONFERENCE (Konference, Otazky) 1:N

DOPRAVA_KONFERENCE (Konference, Doprava) 1:N

PRIHLASENI_DOPRAVA (Prihlaseni, Doprava) 1:N

UZIVATEL_STAT (Uzivatel, Stat) 1:N

PROGRAM_KONFERENCE (Konference, Program) 1:N

PROGRAM_OBSAHUJE_UDALOSTI (Program, Udalosti) 1:N

SBORNIK_KONFERENCE (Konference, Sbornik) 1:N

SBORNIK_OBSAHUJE_CLANKY (Sbornik, Clanek) 1:N

TEMATA_KONFERENCE (Konference, Temata) 1:N

PRIHLASENI_KE_KONFERENCI (Konference, Prihlaseni) 1:N

UZIVATEL_PRIHLASENI (Uzivatel, Prihlaseni) 1:N

PRIHLASENI_CLANEK (Prihlaseni, Clanek) 1:N

UZIVATEL_V_ROLI (Role, Uzivatel) 1:N

UZIVATELUV_UCET (Uzivatel, Ucet) 1:N

UCET_OBSAHUJE_POLOZKY (Ucet, Polozka) 1:N

UZIVATEL_RECENZUJE (Uzivatel, Hodnoceni) 1:N

CLANEK_MA_RECENZI (Clanek, Hodnoceni) 1:N

TEMATA_CLANKU (Temata, Clanek) 1:N

Příloha č. 4: Datový slovník

Uživatel					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_uzivatele</i>	int	50	PK	N	Y
<i>login</i>	varchar	250	N	N	N
<i>heslo</i>	varchar	250	N	N	N
<i>jmeno</i>	varchar	250	N	N	N
<i>prijmeni</i>	varchar	250	N	N	N
<i>id_role</i>	int	50	FK	N	Y
<i>email</i>	varchar	250	N	N	N
<i>organizace</i>	varchar	250	N	N	N
<i>titul_pred</i>	varchar	50	N	Y	N
<i>titul_za</i>	varchar	50	N	Y	N
<i>pracovni_pozice</i>	varchar	250	N	Y	N
<i>ulice</i>	varchar	250	N	Y	N
<i>město</i>	varchar	250	N	Y	N
<i>PSC</i>	varchar	250	N	Y	N
<i>id_statu</i>	int	50	FK	N	Y
<i>telefon</i>	varchar	250	N	Y	N
<i>fax</i>	varchar	250	N	Y	N
<i>jmenoLogin</i>	varchar	200	N	N	N

Článek					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_clanku</i>	int	50	PK	N	Y
<i>nazev</i>	varchar	500	N	N	N
<i>abstrakt</i>	varchar	1000	N	N	N
<i>odkaz</i>	varchar	500	N	N	N
<i>stav</i>	varchar	50	N	N	N
<i>datum_odeslani</i>	datetime		N	N	N
<i>id_tematu</i>	int		FK	N	Y
<i>zpusob_prezentace</i>	varchar	50	N	N	N
<i>final</i>	varchar	500	N	Y	N
<i>zpusob_uverejneni</i>	varchar	50	N	N	N
<i>id_sborniku</i>	int	50	FK	N	Y
<i>id_prihlaseni</i>	int	50	FK	N	Y

Konference					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_konference</i>	int	50	PK	N	Y
<i>aktivni_konference</i>	bit	2	N	Y	N
<i>nazev_konference</i>	varchar	500	N	N	N
<i>poznámka</i>	varchar	2000	N	Y	N

Hodnoceni					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_hodnoceni</i>	int	50	PK	N	Y
<i>id_uzivatele</i>	int	50	FK	N	Y
<i>id_clanku</i>	int	50	FK	N	Y
<i>originality</i>	int	5	N	Y	N
<i>originality_comment</i>	varchar	200	N	Y	N
<i>novelty</i>	int	5	N	Y	N
<i>novelty_comment</i>	varchar	200	N	Y	N
<i>innovation</i>	int	5	N	Y	N
<i>innovation_comment</i>	varchar	200	N	Y	N
<i>relevance</i>	int	5	N	Y	N
<i>relevance_comment</i>	varchar	200	N	Y	N
<i>appropriateness</i>	int	5	N	Y	N
<i>appropriateness_comment</i>	varchar	200	N	Y	N
<i>significance</i>	int	5	N	Y	N
<i>significance_comment</i>	varchar	200	N	Y	N
<i>quality</i>	int	5	N	Y	N
<i>quality_comment</i>	varchar	200	N	Y	N
<i>presentation</i>	varchar	5	N	Y	N
<i>presentation_comment</i>	varchar	200	N	Y	N
<i>recommendation</i>	varchar	200	N	Y	N
<i>recommendation_comment</i>	varchar	200	N	Y	N
<i>degree</i>	int	50	N	Y	N
<i>comment</i>	varchar	500	N	Y	N
<i>stav_recenze</i>	varchar	50	N	Y	N

Udalosti					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_udalosti</i>	int	50	PK	N	Y
<i>id_programu</i>	int	50	FK	N	Y
<i>start_udalosti</i>	datetime		N	Y	N
<i>konec_udalosti</i>	datetime		N	Y	N
<i>popis_udalosti</i>	varchar	500	N	Y	N
<i>priorita</i>	int	50	N	Y	N

Temata					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_tematu</i>	int	50	PK	N	Y
<i>tema</i>	varchar	250	N	N	N
<i>id_konference</i>	int	50	FK	N	Y

Prihlaseni					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_prihlaseni</i>	int	50	PK	N	Y
<i>id_uzivatele</i>	int	50	FK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>ucast_na_konferenci</i>	bit	2	N	Y	N
<i>id_dopravy</i>	int	50	FK	N	Y

Ucet					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_uctu</i>	int	50	PK	N	Y
<i>id_uzivatele</i>	int	50	FK	N	Y
<i>stav_uctu</i>	int	50	N	N	N
<i>zaplaceno</i>	bit	2	N	Y	N

Polozka					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_polozky</i>	int	50	PK	N	Y
<i>id_uctu</i>	int	50	FK	N	Y
<i>druh_polozky</i>	varchar	50	N	Y	N
<i>cena_polozky</i>	int	50	N	Y	N

Role					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_role</i>	int	50	PK	N	Y
<i>nazev_role</i>	varchar	50	N	N	N

Sbornik					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_sborniku</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>popis</i>	varchar	500	N	Y	N

Program					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_programu</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>popis</i>	varchar	500	N	Y	N

Stat					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_statu</i>	int	50	PK	N	Y
<i>nazev</i>	varchar	50	N	N	N

Doprava					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_dopravy</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>druh_dopravy</i>	varchar	100	N	N	N
<i>mista_v_buse</i>	bit	2	N	Y	N

Otazky					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_otazky</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>nazev_otazky</i>	varchar	50	N	N	N
<i>text_otazky</i>	varchar	2000	N	N	N

Page					
<i>název</i>	typ	velikost	klíč	null	index
<i>název</i>	varchar	50	PK	N	Y
<i>obsah</i>	varchar	MAX	N	Y	N
<i>id_konference</i>	int	50	FK	N	Y

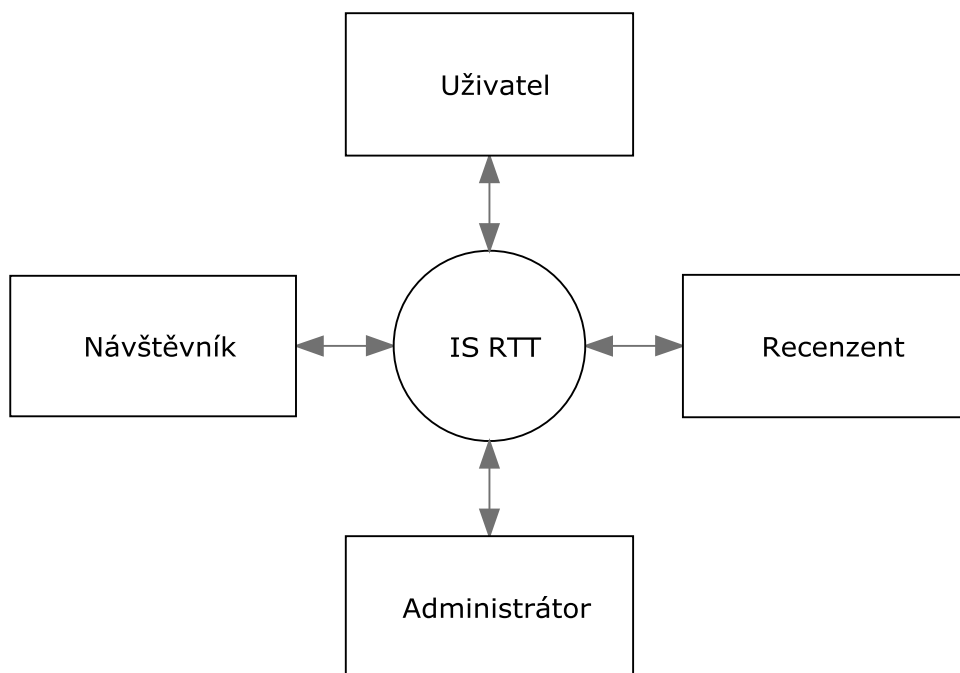
Termíny					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_terminu</i>	int	50	PK	N	Y
<i>popis_terminu</i>	varchar	500	N	Y	N
<i>termin</i>	datetime		N	Y	N
<i>id_konference</i>	int	50	FK	N	Y

Nastavení					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_nastaveni</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>smtp_ip</i>	varchar	50	N	N	N
<i>mist_v_autobuse</i>	int	50	N	Y	N
<i>mist_v_hotelu</i>	int	50	N	Y	N

Emaily					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_emailu</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>název_emailu</i>	varchar	500	N	Y	N
<i>subject_emailu</i>	varchar	500	N	Y	N
<i>telo_emailu</i>	varchar	MAX	N	Y	N
<i>from_emailu</i>	varchar	500	N	Y	N
<i>display_name_emailu</i>	varchar	500	N	Y	N
<i>isHtml</i>	bit		N	Y	N

Stavy					
<i>název</i>	typ	velikost	klíč	null	index
<i>id_stavu</i>	int	50	PK	N	Y
<i>id_konference</i>	int	50	FK	N	Y
<i>popis_stavu</i>	varchar	50	N	N	N
<i>barva_stavu</i>	varchar	50	N	N	N

Příloha č. 5: Kontextový diagram



Příloha č. 6: SCRUM – Na přiloženém DVD se nachází dokument obsahující veškerou agendu spjatou s touto metodikou. Nachází se zde kompletní Product Backlog, všechny Sprint Backlogy a k nim patřící Burn-Down Charty. Jsou zde uvedeny i kompletní výpočty velocity týmu a jednotlivých vývojářů.